

ADI splitting schemes for a fourth-order nonlinear partial differential equation from image processing

Article (Accepted Version)

Calatroni, Luca, Düring, Bertram and Schönlieb, Carola-Bibiane (2014) ADI splitting schemes for a fourth-order nonlinear partial differential equation from image processing. *Discrete and Continuous Dynamical Systems - Series A*, 34 (3). pp. 931-957. ISSN 1078-0947

This version is available from Sussex Research Online: <http://sro.sussex.ac.uk/id/eprint/59615/>

This document is made available in accordance with publisher policies and may differ from the published version or from the version of record. If you wish to cite this item you are advised to consult the publisher's version. Please see the URL above for details on accessing the published version.

Copyright and reuse:

Sussex Research Online is a digital repository of the research output of the University.

Copyright and all moral rights to the version of the paper presented here belong to the individual author(s) and/or other copyright owners. To the extent reasonable and practicable, the material made available in SRO has been checked for eligibility before being made available.

Copies of full text items generally can be reproduced, displayed or performed and given to third parties in any format or medium for personal research or study, educational, or not-for-profit purposes without prior permission or charge, provided that the authors, title and full bibliographic details are credited, a hyperlink and/or URL is given for the original metadata page and the content is not changed in any way.

ADI SPLITTING SCHEMES FOR A FOURTH-ORDER NONLINEAR PARTIAL DIFFERENTIAL EQUATION FROM IMAGE PROCESSING

LUCA CALATRONI

Cambridge Centre for Analysis, University of Cambridge
Wilberforce Road, CB3 0WA, Cambridge, United Kingdom
(lc524@cam.ac.uk)

BERTRAM DÜRING

Department of Mathematics, University of Sussex
Pevensey II, BN1 9QH, Brighton, United Kingdom
(b.during@sussex.ac.uk)

CAROLA-BIBIANE SCHÖNLIEB

Department of Applied Mathematics and Theoretical Physics (DAMTP)
University of Cambridge, Wilberforce Road, CB3 0WA, Cambridge, United Kingdom
(cbs31@cam.ac.uk)

Dedicated to Prof. Arie Iserles, on the occasion of his 65th birthday.

Abstract

We present directional operator splitting schemes for the numerical solution of a fourth-order, nonlinear partial differential evolution equation which arises in image processing. This equation constitutes the H^{-1} -gradient flow of the total variation and represents a prototype of higher-order equations of similar type which are popular in imaging for denoising, deblurring and inpainting problems. The efficient numerical solution of this equation is very challenging due to the stiffness of most numerical schemes. We show that the combination of directional splitting schemes with implicit time-stepping provides a stable and computationally cheap numerical realisation of the equation.

Key words: ADI splitting, higher-order nonlinear diffusion, total variation, image processing.

1 Introduction

In this paper we propose directional operator splitting methods for the numerical solution of fourth-order nonlinear partial differential equations of the type

$$(1.1) \quad \begin{aligned} u_t &= \nabla \cdot (h(u) \nabla q) & q &\in \partial \mathcal{E}(u) & \text{in } \Omega \times (0, \infty), \\ u(t=0) &= u_0 & & & \text{in } \Omega, \end{aligned}$$

where \mathcal{E} is the total variation (TV) seminorm (see, for instance, [1])

$$(1.2) \quad \mathcal{E}(u) := |Du|(\Omega) = \sup_{\mathbf{p} \in C_0^\infty(\Omega; \mathbb{R}^2), \|\mathbf{p}\| \leq 1} \int_{\Omega} u \nabla \cdot \mathbf{p} \, dx.$$

and $\partial\mathcal{E}u$ is the subdifferential of \mathcal{E} in u (see [1]). Here $\Omega \subset \mathbb{R}^2$ is open and bounded with Lipschitz boundary, $h : \mathbb{R} \rightarrow \mathbb{R}$, and u_0 a sufficiently regular initial condition. In our considerations equation (1.1) is endowed with periodic boundary conditions. The elements q of the subdifferential $\partial\mathcal{E}$ have the property that, if $q \in \partial\mathcal{E}(u)$, then (see [52, Proposition 4.1]):

$$(1.3) \quad q = \begin{cases} -\nabla \cdot \left(\frac{\nabla u}{|\nabla u|} \right) & \text{if } |\nabla u(x)| \neq 0, \\ 0 & \text{if } |\nabla u(x)| = 0. \end{cases}$$

The above characterization shows the fourth-differential order and the strong nonlinearity in equation (1.1).

Equations of the form (1.1) typically arise in the study of tension driven flow of thin liquid films (see [41, 43]), and have recently found applications in image processing (see, e.g., [44, 8, 49]). In this paper we limit ourselves to the consideration of one prototype of (1.1), that is we consider the special case where $h(u) \equiv 1$. The equation we obtain in that case reads

$$(1.4) \quad u_t = \Delta q, \quad q \in \partial\mathcal{E}(u), \quad \text{in } \Omega \times (0, \infty),$$

which constitutes a gradient flow of the total variation seminorm in the space H^{-1} .

In image processing problems – such as image denoising and inpainting – nonlinear partial differential equations of the type (1.4) became popular due to their ability of preserving edges in the process of reconstruction (see [46, 11, 47] for instance). In the latter works the authors typically deal with second-order partial differential equations, that is L^2 -gradient flows of the total variation functional. Instead, in [38, 44, 40, 33, 19] an approach like (1.4) is proposed for image denoising and image decomposition. When applied to image denoising, a given noisy image u_0 is decomposed into its piecewise smooth part $u = u(T)$, solution of (1.4) at time T , and its oscillatory, noisy part n , i.e., $u_0 = u + n$. Similarly, in image decomposition the piecewise smooth part u represents the structure/cartoon part of the image, and the oscillatory part n its texture part. The advantage of using an H^{-1} subgradient of the total variation rather than an L^2 subgradient is that (1.4) shows better separability of the data into oscillatory and piecewise constant components and it reduces unwanted artifacts of the total variation filter such as staircasing, cf. [44, 33, 36].

In [8, 48] equation (1.4) is used for image inpainting, i.e. the problem of filling the missing parts of damaged images using the information from the surrounding areas. Here, the higher-order subgradient is necessary for improving upon the ability of total variation-type inpainting to smoothly connect image structures also over large distances, cf. for instance [14, 15, 16, 12, 37, 25]. In [8] the inpainted image u is reconstructed from a given image $f \in L^2(\Omega)$ which is damaged inside the so-called inpainting domain $D \subset \Omega$ by evolving it via the flow

$$(1.5) \quad \begin{aligned} u_t &= \Delta q, & q &\in \partial TV(u), & \text{in } D, \\ u &= f & & & \text{in } \Omega \setminus D, \end{aligned}$$

where

$$TV(u) := \begin{cases} |Du|(\Omega) & \text{if } \|u\|_\infty \leq 1 \quad \text{a.e. in } \Omega, \\ +\infty & \text{otherwise.} \end{cases}$$

Here, the L^∞ -bound on solutions of (1.5) is a technical assumption needed for the existence analysis in [7] and does not present a restriction when dealing with grayvalue images whose values are always bounded within a fixed interval.

If the function h in (1.1) is the identity function $h(u) = u$ we obtain the following equation

$$(1.6) \quad \begin{aligned} u_t &= \nabla \cdot (u \nabla q), \quad q \in \partial \mathcal{E}(u), \quad \text{in } \Omega \times (0, \infty), \\ u(0, x) &= u_0(x) \geq 0 \quad \text{in } \Omega. \end{aligned}$$

This equation can be formally derived as the L^2 -Wasserstein gradient flow of the total variation \mathcal{E} in (1.2) and has been proposed in [7] for density estimation and smoothing. Equation (1.6) has been further investigated in [20], where the authors numerically studied the scale space properties and high-contrasting effects of the equation by exploiting a numerical scheme similar to the ones we consider in this paper. In [6] the authors consider such equation for the problem of denoising, solving an alternative primal-dual formulation of (1.6) with a Newton-type method.

From a computational point of view, finding numerical schemes that solve equations of the form (1.1) is a challenging problem. Dealing with an evolutionary nonlinear fourth-order partial differential equation, we would like to find an efficient and easily applicable method. Note that a naive explicit discretisation in time may restrict the choice of the time-step size Δt up to an order Δx^4 , where Δx denotes the step size of the spatial grid, compare [51]. Moreover, the strong nonlinearity of subgradients of the total variation adds additionally constraints to the stability condition of a discrete time stepping scheme, compare [13, 19]. In particular, when approximating the subgradient of the total variation by regularising it – either by a square root ε -regularisation or by a regularisation of its dual formulation – the size of the regularisation parameter encodes the accuracy of this approximation, that is the strength of the nonlinearity in the approximated subgradient. The presence of this nonlinearity together with the fourth differential order of the equation then may result in restrictive stability conditions on numerical time stepping for small values of this regularisation parameter.

Having these complications in mind, some numerical methods have been proposed to solve equations like (1.4). Lieu and Vese [33] proposed a numerical method to solve TV- H^{-1} denoising/decomposition by using the Fourier representation of the H^{-1} norm on the whole \mathbb{R}^d , $d \geq 1$. This leads to a second-order PDE defined in Fourier space. In [23] and [24] the authors propose an algorithm using a finite element method to solve such equation, while in [2, 50] a dual approach similar to the one described in [10] is presented with interesting applications both to denoising and inpainting. In [48] the authors present results of convergence and unconditional stability for a particular numerical splitting method solving equation (1.4), called *convexity splitting*. Therein, the equation is modeled as the gradient flow in H^{-1} of the difference of two convex energies. The result is the presence of a linear diffusion term in the numerical scheme which balances the unstable behaviours coming from the nonlinear terms.

In this work we are interested in performing a *directional* operator splitting for the numerical solution of equation (1.4), i.e. (1.5) for image inpainting. In particular, we consider *alternating directional splitting (ADI)* methods that reach back to [45] and have been further developed in, e.g. [32, 31, 4, 28, 29, 55]. We discuss three different splitting methods and their stability properties. Our investigation culminates in the proposal of a fully-implicit, additive multiplicative operator splitting scheme for a slightly modified form of equation (1.4). In our numerical experiments we obtain stable numerical solutions for arbitrary values of the regularization parameter, suggesting unconditional stability of the method. This allows to use fewer iterations (larger time steps) while each iteration is still computationally cheap due to the directional splitting.

We want to stress, that in the existing literature so far, most methods have been considered for the solution of second-order linear and nonlinear PDEs. Up to our knowledge the following discussion is the first one that carries out an ADI splitting for a higher-order total variation flow, which is both highly nonlinear and of fourth differential order.

Outline of the paper. After the preliminary Section 2, in Section 3 we will review the general features of the main ADI methods. We will apply them to equation (1.4) in Section 4. In order to come up with new and efficient ADI methods solving numerically (1.4), we exploit at first (see Section 4.1) an auxiliary, linear, fourth-order equation. We next present in Section 4.2 our ideas applied to the nonlinear cases we are interested on. Our strategy is as follows: after linearising the equation in a suitable way, we expand and discretise the appearing differential operators and we build up the ADI schemes where we alternatively apply differential operators acting just along either the x - or the y -direction. According to the different choices of linearisation performed, some mixed derivative terms may appear as well. They need to be controlled properly in order to get stable results. Stability properties appear to be related also with the regularisation of the term $|\nabla u|$ appearing in both equations. In Sections 4.2.1 and 4.2.2 we will give more details on that, presenting in Section 4.2.4 a modified stable ADI scheme dealing implicitly just with pure derivatives. Section 4.3 presents a quite different approach to solve the so-called *primal-dual* formulation of the TV- H^{-1} problem. In the concluding Section 5 we present some numerical results obtained by applying the ADI methods above to some problems arising in image processing.

2 Preliminaries

In this section we briefly introduce the main notation employed in the paper and the definition of the differential discrete operators we will use in the following to describe the numerical schemes. Dealing with a fourth-order nonlinear PDE, we also need to make precise what we mean exactly by *stability* of a numerical scheme.

2.1 Notation

In this paper we discuss the numerical solution of evolutionary differential equations. We need to distinguish between the exact solution u of the continuous equation and the approximate semi-discrete and fully discrete solutions of the numerical schemes. Let

$u(x, y, t)$ the exact solution at point $(x, y) \in \Omega = [a, b] \times [c, d]$ and $t \geq 0$. We write $u_n(x, y)$ to indicate the approximation in time of the function $u(x, y, t_n)$, where $(x, y) \in \Omega$, $t_n = n\Delta t$, $n \geq 0$ and Δt is the time step size. Further, we approximate Ω by a finite grid $\{a = x_1 < \dots < x_N = b\} \times \{c = y_1 < \dots < y_M = d\}$ with equidistant spatial size $h = (b - a)/N = (d - c)/M$. We then denote by $U(t)$ the approximation of $u(x_i, y_j, t)$ in the node (x_i, y_j) at time t . Finally, the fully discrete approximation of u is denoted by $U = (U_n)_{i,j}$. When dealing with vectors, we will indicate their components using the superscripts notation: $\mathbf{Y} = (Y^1 Y^2)^\top$. Finally, we will indicate by δ_* the differential operator acting in the direction $*$ and we will use the notations D_∇ , D_{div} and $D_\Delta U$ to indicate the discrete gradient, divergence and laplacian of the approximating solution U , respectively.

2.2 Definition of the discrete operators

In the following we define all the discrete operators that we use for approximating the spatial derivatives that appear in our numerical schemes. We use periodic boundary conditions throughout. We also performed the discretisation changing the boundary conditions to homogeneous Neumann boundary conditions, but this did not affect the numerical results significantly.

We approximate the first derivatives $u_x(x_i, y_j, \cdot)$ by forward differences $(\delta_x^+ U)_{i,j}$ and backward differences $(\delta_x^- U)_{i,j}$, where:

$$(\delta_x^+ U)_{i,j} = \frac{u_{i+1,j} - u_{i,j}}{h} \quad \text{and} \quad (\delta_x^- U)_{i,j} = \frac{u_{i,j} - u_{i-1,j}}{h}$$

The first derivatives of u with respect to y are approximated analogously by $(\delta_y^+ U)_{i,j}$ and by $(\delta_y^- U)_{i,j}$. The pure second derivatives will be approximated by using the five-point formula, this means the Laplacian operator $\Delta u = u_{xx} + u_{yy}$ is approximated by:

$$(D_\Delta U)_{i,j} = (\delta_{xx} U)_{i,j} + (\delta_{yy} U)_{i,j} = \frac{u_{i+1,j} + u_{i-1,j} + u_{i,j+1} + u_{i,j-1} - 4u_{i,j}}{h^2}.$$

The mixed derivatives $u_{xy}(x_i, y_j, \cdot)$ are approximated by:

$$(\delta_{xy} U)_{i,j} = \frac{u_{i+1,j+1} + u_{i-1,j-1} - u_{i-1,j+1} - u_{i+1,j-1}}{4h^2}.$$

For the discretisation of the pure and mixed fourth-order derivatives appearing in Section 4.1, we will use a 5×5 stencil and approximate by:

$$\begin{aligned} (\delta_{xxxx} U)_{i,j} &= (\delta_{xx}(\delta_{xx} U))_{i,j} = \frac{u_{i+2,j} - 4u_{i+1,j} + 6u_{i,j} - 4u_{i-1,j} + u_{i-2,j}}{h^4}, \\ (\delta_{yyyy} U)_{i,j} &= (\delta_{yy}(\delta_{yy} U))_{i,j} = \frac{u_{i,j+2} - 4u_{i,j+1} + 6u_{i,j} - 4u_{i,j-1} + u_{i,j-2}}{h^4}, \\ (\delta_{xxyy} U)_{i,j} &= (\delta_{xy}(\delta_{xy} U))_{i,j} = \frac{u_{i+2,j+2} + u_{i-2,j-2} + 4u_{i,j} + u_{i-2,j+2} + u_{i+2,j-2}}{16h^4} \\ &\quad - \frac{u_{i,j+2} + u_{i+2,j} + u_{i-2,j} + u_{i,j-2}}{8h^4}. \end{aligned}$$

2.3 Definition of stability

In the following, we introduce the notion of *stability* of the numerical schemes solving the nonlinear equations (1.4) and (1.5):

Definition 2.1. *Let u be an element of a suitable function space \mathcal{H} defined on $\Omega \times [0, T]$, with $\Omega \subset \mathbb{R}^2$ open and bounded and $T > 0$. Let F be a real valued function and $u_t = G(u, D^\alpha u)$ a partial differential equation with all spatial D^α with order $|\alpha| \leq 4$. A corresponding time stepping method*

$$(2.7) \quad u_{n+1} = u_n + \Delta t G_n(u_n, u_{n+1}, D^\alpha u_n, D^\alpha u_{n+1})$$

where G_n is a suitable approximation of G in u_n and u_{n+1} is **unconditionally stable** if all the solutions of (2.7) are bounded for all $\Delta t > 0$ and all n such that $n\Delta t \leq T$ and **stable** if, for a given $\Delta t > 0$, the solutions of (2.7) are bounded for all n such that $n\Delta t \leq T$.

3 ADI splitting schemes

In this section we introduce the numerical method of directional splitting. In particular, we discuss three types of directional splitting: the Peaceman-Rachford scheme, the Douglas-Hundsdorfer scheme, and an additive multiplicative operator splitting scheme. In everything that follows we consider large systems of generic ordinary differential equations that arise from a semi-discretisation of initial boundary value problems such as

$$(3.1) \quad \begin{cases} U'(t) = F(t, U(t)) & \text{for } t \geq 0, \\ U(0) = U_0, \end{cases}$$

for a given function F . The explicit dependence on time of F may not appear, thus considering *autonomous* systems. Typically, directional splitting methods (see, for instance, [31]) decompose the function F into the sum:

$$(3.2) \quad F(t, U) = F_0(t, U) + F_1(t, U) + \cdots + F_s(t, U) \quad \text{for some } s \geq 1,$$

where s is the spatial dimension of the problem. The components F_j , $j = 1, \dots, s$, encode the linear action of F along the space direction $j = 1, \dots, s$, respectively, and F_0 contains contributes coming from mixed directions and/or non stiff nonlinear terms. A generic ADI scheme constitutes a time-stepping method that treats the unidirectional components F_j , $j \geq 1$ implicitly and the F_0 component, if present, explicitly in time.

3.1 The Peaceman-Rachford scheme

The first method we consider in this framework is the second-order *Peaceman-Rachford* ADI method (see [45] and [31]) where the operator F can be splitted into $F = F_1 + F_2$, i.e. no mixed derivative or nonlinear terms are present. Let $\Delta t > 0$ be the time step size

of the scheme, then for every $n \geq 1$ the numerical solution U_{n+1} of (3.1) is computed as follows:

$$(3.3) \quad \begin{cases} U_{n+1/2} = U_n + \frac{\Delta t}{2} F_1(t_n, U_n) + \frac{\Delta t}{2} F_2(t_{n+1/2}, U_{n+1/2}), \\ U_{n+1} = U_{n+1/2} + \frac{\Delta t}{2} F_1(t_{n+1}, U_{n+1}) + \frac{\Delta t}{2} F_2(t_{n+1/2}, U_{n+1/2}). \end{cases}$$

In (3.3) we can see that forward and backward Euler are applied alternatingly in a symmetrical fashion, thus obtaining second-order accuracy (see [31]). In each half time step, one of the two dimensions is treated implicitly, whereas the other one is explicitly considered. Note that (3.3) can be generalised to problems with nonlinearities $f(U)$ and

$$U'(t) = F(t, U(t)) + f(U(t)),$$

by adding $\frac{1}{2}f(U_n)$ and $\frac{1}{2}f(U_{n+1/2})$ to the right hand side of the first and second equation in (3.3) respectively. The scheme (3.3), however, does not have a natural extension for more than two components F_j , so more general ADI methods have been proposed in the literature.

3.2 The Douglas-Hundsdorfer scheme

Another ADI method that allows a more general decomposition like the one in (3.2) is the so-called *Douglas* method (see [30], [31] and [32]). In that, the numerical approximation in each time step is computed by applying at first a forward Euler predictor and then it is stabilised by fractional s steps where just the unidirectional components F_j appear, weighted by a parameter $\theta \in [0, 1]$. Its size controls the implicit/explicit character of such steps. In other words, the unidirectional operators are applied to the convex combination $\theta U_{n+1} + (1 - \theta)U_n$, thus considering fully implicit steps for $\theta = 1$, explicit ones for $\theta = 0$ and a Crank-Nicolson type scheme when $\theta = 1/2$. The consistency order in time of the scheme is equal to two whenever $F_0 = 0$ and $\theta = 1/2$ and it is of order one otherwise. In many applications we need to consider, however, $F_0 \neq 0$ (for instance, if we are considering contributions coming from mixed derivative operators) for any given θ . Some extensions of the Douglas scheme have been proposed in order to overcome this drawback. The following scheme proposed in [32] by Hundsdorfer is an extension of the *Douglas* method where a second stabilising parameter $\sigma > 0$ appears:

$$(3.4) \quad \begin{cases} Y_0 = U_n + \Delta t F(t_n, U_n) \\ Y_j = Y_{j-1} + \theta \Delta t (F_j(t_{n+1}, Y_j) - F_j(t_n, U_n)), \quad j = 1, 2, \dots, s \\ \tilde{Y}_0 = Y_0 + \sigma \Delta t (F(t_{n+1}, Y_s) - F(t_n, U_n)) \\ \tilde{Y}_j = \tilde{Y}_{j-1} + \theta \Delta t (F_j(t_{n+1}, \tilde{Y}_j) - F_j(t_{n+1}, Y_s)), \quad j = 1, 2, \dots, s \\ U_{n+1} = \tilde{Y}_s. \end{cases}$$

The advantage of this extension is that for any given θ the scheme (3.4) has time-consistency order two if $\sigma = 1/2$ and one otherwise, independently of F_0 . The parameter θ is typically fixed to $\theta = 1/2$. Its choice is discussed in [32]. Larger values of θ give stronger damping of implicit terms, lower values typically better accuracy. Stability

properties of this scheme when applied to linear convection-diffusion equations with mixed derivative terms have been investigated in [28, 29]. There, the preferable value for θ is $\theta = 1/2 + \sqrt{3}/6$. In [55] the authors combine the approach presented above with iterative methods for solving nonlinear systems.

3.3 Additive multiplicative operator splitting schemes

As pointed out above, in both schemes (3.3) and (3.4) some explicit terms appear. These may affect stability properties of the methods and, generally, their accuracy. As described in [4], splitting schemes as directional splitting methods belong to the family of *multiplicative locally one-dimensional* (LOD) schemes. In their general semi-implicit form, when a splitting similar to (3.2) holds, they appear as:

$$(3.5) \quad \prod_{i=0}^s (I - \Delta t F_i) U_{n+1} = U_n,$$

where, similarly as before, each operator F_i , $1 \leq i \leq s$, is acting just along the i -th direction, whereas F_0 encodes mixed contributions. As pointed out above, it is generally difficult to deal with such an operator as the matrix $(I - \Delta t F_0)$ is, generally, not tridiagonal. For this reason, the scheme (3.4) deals explicitly with such a term. Analogously, explicit components appear also when applying (3.3) because of the alternating application of forward and backward Euler. As we will point out later on, these explicit contributions may create stability problems in the methods we are going to present. Following the strategy presented in [4], our attempt is to modify (3.3) such that no explicit contributions appear. The cost of such an operation will be reducing the accuracy of the method to order one, against the second-order achieved with the classical Peaceman-Rachford method (3.3). In order to preserve such accuracy as well as the symmetry of the method, at each time step two calculations are performed:

$$(3.6) \quad \begin{cases} (I - \Delta t F_1) U_{n^*} = U_n \\ (I - \Delta t F_2) \tilde{U}_{n+1} = U_{n^*} \end{cases} \quad \text{and} \quad \begin{cases} (I - \Delta t F_2) U_{n^*} = U_n \\ (I - \Delta t F_1) \bar{U}_{n+1} = U_{n^*} \end{cases}$$

which, written in the same form as (3.3) and (3.4), read as:

$$\begin{cases} U_{n^*} = U_n + \Delta t F_1(U_{n^*}) \\ \tilde{U}_{n+1} = U_{n^*} + \Delta t F_2(\tilde{U}_{n+1}) \end{cases} \quad \text{and} \quad \begin{cases} U_{n^*} = U_n + \Delta t F_2(U_{n^*}) \\ \bar{U}_{n+1} = U_{n^*} + \Delta t F_1(\bar{U}_{n+1}) \end{cases}.$$

To get the numerical solution U_{n+1} we simply average:

$$(3.7) \quad U_{n+1} = \frac{\tilde{U}_{n+1} + \bar{U}_{n+1}}{2},$$

thus ensuring a symmetric splitting. Due to the nature of such a method, we refer to (3.6)-(3.7) as *additive multiplicative operator splitting* (AMOS) ADI method. Note, that this scheme is identical to an earlier version of the well-known *Strang* splitting (see, for instance, [31, (1.12), p.329]). For nonlinear problems, such a scheme is second-order accurate, in contrast to first-order accuracy of the classical *Strang* splitting scheme (compare [4]). Furthermore, the scheme (3.6)-(3.7) has the advantage of allowing a parallel implementation, as suggested in [34, 35, 54].

4 Applications to higher-order PDEs

As we are dealing with the particular case of regular domains in \mathbb{R}^2 , we will consider in the following $s = 2$. Thus, the F_1 component (F_2 , respectively) will contain operators acting just along the x -direction (y -direction, respectively). When appearing, the term F_0 will deal, instead, with the mixed xy -direction. Our aim is to adapt the ADI schemes (3.3), (3.4) and (3.6)-(3.7) to regularised versions of equation (1.4) in Sections 4.2 and 4.3.

4.1 A linear example: the biharmonic equation

We start by considering the auxiliary linear, fourth-order parabolic *biharmonic* equation:

$$(4.1) \quad u_t = -\Delta^2 u \quad \text{in } \Omega \times (0, T).$$

Such equation appears in many applied mathematical models such as the Cahn-Hilliard models describing phase transitions and separation in binary mixtures (see, for instance, [22] and [42]). Some work on ADI schemes applied to the biharmonic equation already exists in literature. It dates back to [18] where the authors consider the equation to model vibrational modes for thin plates and it has been analysed also in [55, 28] where linear stability results are proved as well. We are looking for the solution U of the following semi-discretised version of (4.1):

$$(4.2) \quad U_t = F(U) := -(D_{\Delta^2})U = -\delta_{xxxx}U - \delta_{yyyy}U - 2\delta_{xxyy}U.$$

According to the rules of ADI schemes, we decompose the function F into the sum

$$F(U) = F_0(U) + F_1(U) + F_2(U)$$

where the components F_i , $i = 0, 1, 2$, are defined by

$$F_0(u) := -2\delta_{xxyy}U, \quad F_1(u) := -\delta_{xxxx}U, \quad F_2(u) := -\delta_{yyyy}U,$$

and the differential operators have been discretised as discussed in Section 2. With this choice, we can find for every n the approximating solution U_{n+1} of (4.1) using the *Hundsdorfer* ADI scheme (3.4).

Simplifying the problem by splitting the fourth-order equation into a mathematically equivalent autonomous system of two partial differential equations of order two produces the system:

$$(4.3) \quad \begin{cases} U_t = D_{\Delta}V = \delta_{xx}V + \delta_{yy}V = F(U, V), \\ V = -D_{\Delta}U = -\delta_{xx}U - \delta_{yy}U = G(U, V). \end{cases}$$

Then, the *Hundsdorfer* scheme applied to (4.2) can be equivalently written as a coupled ADI scheme for approximate solutions (U_n, V_n) of (4.3). For positive parameters θ, σ this

gives:

$$(4.4) \quad \left\{ \begin{array}{l} \begin{pmatrix} Y_0^2 \\ Y_0^1 \end{pmatrix} = \begin{pmatrix} G(U_n, V_n) \\ U_n + \Delta t F(U_n, Y_0^2) \end{pmatrix}, \\ \begin{pmatrix} Y_1^1 \\ Y_1^2 \end{pmatrix} = \begin{pmatrix} Y_0^1 \\ 0 \end{pmatrix} + \begin{pmatrix} \theta \Delta t F_1(Y_1^1, Y_1^2) \\ G_1(Y_1^1, Y_1^2) - G_1(U_n, V_n) \end{pmatrix}, \\ \begin{pmatrix} Y_2^1 \\ Y_2^2 \end{pmatrix} = \begin{pmatrix} Y_1^1 \\ 0 \end{pmatrix} + \begin{pmatrix} \theta \Delta t F_2(Y_2^1, Y_2^2) \\ G_2(Y_2^1, Y_2^2) - G_2(U_n, V_n) \end{pmatrix}, \\ \begin{pmatrix} \tilde{Y}_0^2 \\ \tilde{Y}_0^1 \end{pmatrix} = \begin{pmatrix} G(Y_2^1, Y_2^2) \\ Y_0^1 + \sigma \Delta t (F(Y_2^1, \tilde{Y}_0^2) - F(U_n, V_n)) \end{pmatrix}, \\ \begin{pmatrix} \tilde{Y}_1^1 \\ \tilde{Y}_1^2 \end{pmatrix} = \begin{pmatrix} \tilde{Y}_0^1 \\ 0 \end{pmatrix} + \begin{pmatrix} \theta \Delta t F_1(\tilde{Y}_1^1, \tilde{Y}_1^2) \\ G_1(\tilde{Y}_1^1, \tilde{Y}_1^2) - G_1(Y_2^1, Y_2^2) \end{pmatrix}, \\ \begin{pmatrix} U_{n+1} \\ V_{n+1} \end{pmatrix} = \begin{pmatrix} \tilde{Y}_1^1 \\ 0 \end{pmatrix} + \begin{pmatrix} \theta \Delta t F_2(\tilde{Y}_2^1, \tilde{Y}_2^2) \\ G_2(\tilde{Y}_2^1, \tilde{Y}_2^2) - G_2(Y_2^1, Y_2^2) \end{pmatrix} \end{array} \right.$$

where the functions F, F_1, F_2 and G, G_1, G_2 are given by:

$$(4.5) \quad \begin{aligned} \begin{pmatrix} F_1(U, V) \\ G_1(U, V) \end{pmatrix} &= \begin{pmatrix} A_1 & B_1 \\ C_1 & D_1 \end{pmatrix} \cdot \begin{pmatrix} U \\ V \end{pmatrix} = \begin{pmatrix} 0 & \delta_{xx} \\ -\delta_{xx} & 0 \end{pmatrix} \cdot \begin{pmatrix} U \\ V \end{pmatrix}, \\ \begin{pmatrix} F_2(U, V) \\ G_2(U, V) \end{pmatrix} &= \begin{pmatrix} A_2 & B_2 \\ C_2 & D_2 \end{pmatrix} \cdot \begin{pmatrix} U \\ V \end{pmatrix} = \begin{pmatrix} 0 & \delta_{yy} \\ -\delta_{yy} & 0 \end{pmatrix} \cdot \begin{pmatrix} U \\ V \end{pmatrix}, \\ F(U, V) &= F_1(U, V) + F_2(U, V), \quad G(U, V) = G_1(U, V) + G_2(U, V). \end{aligned}$$

To verify that (4.4) gives the same solution as the *Hundsdorfer* scheme applied to (4.2) we consider as example the first implicit step in (4.4) providing the approximations (Y_1^1, Y_1^2) . For the approximation Y_1^1 of U_{n+1} we have:

$$Y_1^1 = Y_0^1 + \theta \Delta t F_1(Y_1^1, Y_1^2) = Y_0^1 + \theta \Delta t \delta_{xx}(Y_1^2).$$

Using now the expression of Y_1^2 given by the implicit step relating to the equation for V we have:

$$Y_1^1 = Y_0^1 + \theta \Delta t (\delta_{xx}(-\delta_{xx}(Y_1^1) + \delta_{xx}(U_n))) = Y_0^1 + \theta \Delta t (-\delta_{xxxx}(Y_1^1) + \delta_{xxxx}(U_n)),$$

which, compared to the respective step performed applying the *Hundsdorfer* scheme (3.4) to equation (4.2), gives exactly the same result. We perform the same technique for the other implicit steps. Moreover, as the reader may note, in both the explicit steps of the scheme above we swapped the order of application of the method for consistency issues. Namely, we first find consistent approximations for V_{n+1} using them to get consistent approximations of U_{n+1} . In the application of both these methods we get stable solutions both for $\Delta t = C(\Delta x)^3$ and also for $\Delta t = C(\Delta x)^2$, improving largely upon the condition $\Delta t = C(\Delta x)^4$ that a naive explicit discretisation would give. Scheme (4.4) is indeed *unconditionally* stable, as proved earlier in [28] and confirmed in our numerical experiments in Section 5. However, note that considering bigger time steps such as $\Delta t = C(\Delta x)$ the numerical accuracy suffers not producing sensible solutions.

4.2 Applications to the primal formulation of TV- H^{-1} equation

We now aim to derive an ADI method solving the TV- H^{-1} equation (1.4). Expanding directly the differential operators appearing in the equation generates an intractable number of nonlinear terms of various differential orders. This makes a direct application of the ADI scheme to (1.4) impractical. Therefore, following the ideas presented in Section 4.1, we reduce the original fourth-order equation to an autonomous system of two second-order equations. In the following we consider the *primal* formulation of (1.4), in contrast to the *primal-dual* formulation presented in Section 4.3. To do so, we first regularise the subgradient of the total variation in (1.4) characterised by (1.3). We use the standard *square root ε -regularisation* that replaces $|\nabla u|$ by $|\nabla u|_\varepsilon := \sqrt{|\nabla u|^2 + \varepsilon}$, $1 \gg \varepsilon > 0$, see for instance [9]. This results in the following regularised version of (1.4)

$$(4.6) \quad u_t = -\Delta \nabla \cdot \left(\frac{\nabla u}{|\nabla u|_\varepsilon} \right) \quad \text{and, equivalently,} \quad \begin{cases} u_t = \Delta v, \\ v = -\nabla \cdot \left(\frac{\nabla u}{|\nabla u|_\varepsilon} \right). \end{cases}$$

In the following we present two different linearisations of the problem above. Heuristically, such a choice is important from two different points of view, intrinsically related to each other. The former is the accuracy of the scheme we are considering: rough linearisations (i.e. linearisations which consider most of the nonlinear terms explicitly evaluating them in one or more given approximations of the solution in previous time steps) are likely to present poor accuracy as well as stability issues. This is a general consideration in the numerical solution of every partial differential equation and it has to be taken into account and balanced with the choice of linearisation which might be more accurate and precise, but which could present, on the other hand, difficulties in its implementation and application. The latter point of view is, in some sense, peculiar to our choice of performing a directional splitting scheme. As pointed out above, our purpose is splitting our partial differential operator into the sum of components which are considered both explicitly (see F_0 above) and implicitly (see F_1 and F_2), as in (3.2). As we are going to present in the following, the choice of the linearisation affects such a splitting as the F_0 component and the linearised quantities multiplying the differential operators acting in x and y may change accordingly. For instance, the F_0 component might not appear changing the choice of the ADI scheme we want to use.

In the following we proceed by presenting two ADI schemes of the form (3.3) and (3.4) for two different linearisations of (4.6). For a given initial condition (U_0, V_0) , our problem consists in finding an approximation (U_{n+1}, V_{n+1}) of the solution $(u(t_{n+1}), v(t_{n+1}))$ to (4.6) for every $n \geq 0$. In the following we always linearise around the solution at the previous time step (U_n, V_n) .

4.2.1 The first linearisation

Indicating by \tilde{U} the value of the solution U_n in the previous time step, our first choice of linearisation is the following (compare with [20]):

$$(4.7) \quad \begin{cases} U_t = D_\Delta V, \\ V = -\frac{\varepsilon + (\delta_y^+ \tilde{U})^2}{|D_\nabla^+ \tilde{U}|_\varepsilon^3} \delta_{xx} U - \frac{\varepsilon + (\delta_x^+ \tilde{U})^2}{|D_\nabla^+ \tilde{U}|_\varepsilon^3} \delta_{yy} U + 2 \frac{\delta_x^+ \tilde{U} \delta_y^+ \tilde{U}}{|D_\nabla^+ \tilde{U}|_\varepsilon^3} \delta_{xy} U, \end{cases}$$

where (U, V) is the semi-discrete approximation to a solution of (4.6). Here, the linearisation of V constitutes a semi-implicit approximation of the second-order nonlinear diffusion, evaluating all the first-order derivatives of the expansion in the previous time step. As before, we use the following notation for the system (4.7):

$$(4.8) \quad \begin{pmatrix} U_t \\ V \end{pmatrix} = \begin{pmatrix} F(U, V) \\ G(U, V) \end{pmatrix} = \begin{pmatrix} A & B \\ C & D \end{pmatrix} \cdot \begin{pmatrix} U \\ V \end{pmatrix}$$

for suitable matrices A, B, C and D in $\mathbb{R}^{NM \times NM}$. We split F and G into the sum of three different terms: F_0 and G_0 containing the mixed derivative term and F_1, G_1 and F_2, G_2 containing the derivatives with respect to x and y only, respectively. This produces the splitting:

$$(4.9) \quad \begin{aligned} F(U, V) &= F_0(U, V) + F_1(U, V) + F_2(U, V), \\ G(U, V) &= G_0(U, V) + G_1(U, V) + G_2(U, V), \end{aligned}$$

with:

$$(4.10) \quad \begin{aligned} \begin{pmatrix} F_0(U, V) \\ G_0(U, V) \end{pmatrix} &= \begin{pmatrix} A_0 & B_0 \\ C_0 & D_0 \end{pmatrix} \cdot \begin{pmatrix} U \\ V \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ 2 \frac{\delta_x^+ \tilde{U} \delta_y^+ \tilde{U}}{|D_\nabla^+ \tilde{U}|_\varepsilon^3} \delta_{xy} & 0 \end{pmatrix} \cdot \begin{pmatrix} U \\ V \end{pmatrix}, \\ \begin{pmatrix} F_1(U, V) \\ G_1(U, V) \end{pmatrix} &= \begin{pmatrix} A_1 & B_1 \\ C_1 & D_1 \end{pmatrix} \cdot \begin{pmatrix} U \\ V \end{pmatrix} = \begin{pmatrix} 0 & \delta_{xx} \\ -\frac{\varepsilon + (\delta_y^+ \tilde{U})^2}{|D_\nabla^+ \tilde{U}|_\varepsilon^3} \delta_{xx} & 0 \end{pmatrix} \cdot \begin{pmatrix} U \\ V \end{pmatrix}, \\ \begin{pmatrix} F_2(U, V) \\ G_2(U, V) \end{pmatrix} &= \begin{pmatrix} A_2 & B_2 \\ C_2 & D_2 \end{pmatrix} \cdot \begin{pmatrix} U \\ V \end{pmatrix} = \begin{pmatrix} 0 & \delta_{yy} \\ -\frac{\varepsilon + (\delta_x^+ \tilde{U})^2}{|D_\nabla^+ \tilde{U}|_\varepsilon^3} \delta_{yy} & 0 \end{pmatrix} \cdot \begin{pmatrix} U \\ V \end{pmatrix}. \end{aligned}$$

Due to the presence of a mixed derivative operator, a simple *Peaceman-Rachford* scheme cannot be applied. The *Hundsdoerfer* scheme (3.4) is needed. The ADI scheme we use has the same form as for the biharmonic equation, i.e. we employ (4.4) with F, F_i and G, G_i given by (4.9), (4.10).

4.2.2 The second linearisation

Another possibility is to linearise the system (4.6) in the following way:

$$(4.11) \quad \begin{cases} U_t = D_\Delta V, \\ V = -D_{\text{div}}^- \left(\frac{D_\nabla^+ U}{|D_\nabla^+ \tilde{U}|_\varepsilon} \right) = -\frac{1}{|D_\nabla^+ \tilde{U}|_\varepsilon} \delta_{xx} U + \frac{\delta_x^+ \tilde{U} \delta_{xx} \tilde{U} + \delta_y^+ \tilde{U} \delta_x^- \delta_y^+ \tilde{U}}{|D_\nabla^+ \tilde{U}|_\varepsilon^3} \delta_x^{up} U \\ \quad - \frac{1}{|D_\nabla^+ \tilde{U}|_\varepsilon} \delta_{yy} U + \frac{\delta_x^+ \tilde{U} \delta_x^+ \delta_y^- \tilde{U} + \delta_y^+ \tilde{U} \delta_{yy} \tilde{U}}{|D_\nabla^+ \tilde{U}|_\varepsilon^3} \delta_y^{up} U, \end{cases}$$

where again $\tilde{U} = U_n$ and the spatial quantities are discretised as above and the discrete operators δ_x^{up} and δ_y^{up} are defined below. We observe that with this choice no mixed derivative operator acting on U appears. Mixed terms are encoded and considered in the previous time step. On the other hand, we get first derivative operators and not just second-order ones as in (4.7). Writing again the system (4.11) as in (4.8), we now split F and G in the following way:

$$(4.12) \quad \begin{aligned} F(U, V) &= F_1(U, V) + F_2(U, V), \\ G(U, V) &= G_1(U, V) + G_2(U, V), \end{aligned}$$

where

$$(4.13) \quad \begin{aligned} \begin{pmatrix} F_1(U, V) \\ G_1(U, V) \end{pmatrix} &= \begin{pmatrix} 0 & \delta_{xx} \\ \frac{1}{|D_{\nabla}^+ \tilde{U}|_\varepsilon} \delta_{xx} \frac{\delta_x^+ \tilde{U} \delta_{xx} \tilde{U} + \delta_y^+ \tilde{U} \delta_x^- \delta_y^+ \tilde{U}}{|D_{\nabla}^+ \tilde{U}|_\varepsilon^3} \delta_x^{up} & 0 \end{pmatrix} \cdot \begin{pmatrix} U \\ V \end{pmatrix}, \\ \begin{pmatrix} F_2(U, V) \\ G_2(U, V) \end{pmatrix} &= \begin{pmatrix} 0 & \delta_{yy} \\ -\frac{1}{|D_{\nabla}^+ \tilde{U}|_\varepsilon} \delta_{yy} + \frac{\delta_x^+ \tilde{U} \delta_x^+ \delta_y^- \tilde{U} + \delta_y^+ \tilde{U} \delta_{yy} \tilde{U}}{|D_{\nabla}^+ \tilde{U}|_\varepsilon^3} \delta_y^{up} & 0 \end{pmatrix} \cdot \begin{pmatrix} U \\ V \end{pmatrix}. \end{aligned}$$

We note that the splitting (4.12) is a two-components splitting as the one provided for the *Peaceman-Rachford* ADI scheme (3.3) applied to the system (4.11) and this gives:

$$(4.14) \quad \begin{cases} \begin{pmatrix} U_{n+1/2} \\ V_{n+1/2} \end{pmatrix} = \begin{pmatrix} U_n + \frac{\Delta t}{2} F_1(U_n, V_n) + \frac{\Delta t}{2} F_2(U_{n+1/2}, V_{n+1/2}) \\ G_1(U_n, V_n) + G_2(U_{n+1/2}, V_{n+1/2}) \end{pmatrix}, \\ \begin{pmatrix} U_{n+1} \\ V_{n+1} \end{pmatrix} = \begin{pmatrix} U_{n+1/2} + \frac{\Delta t}{2} F_1(U_{n+1/2}, V_{n+1/2}) + \frac{\Delta t}{2} F_2(U_{n+1/2}, V_{n+1/2}) \\ G_1(U_{n+1/2}, V_{n+1/2}) + G_2(U_{n+1/2}, V_{n+1/2}) \end{pmatrix}. \end{cases}$$

For the discretisation of the first derivative operators in the scheme – present in the equation for V in (4.11) – we use the standard numerical technique of upwinding, i.e. the sign of the coefficients in front of the first derivatives terms affects in which direction the finite differences are computed. More precisely, we use:

$$(4.15) \quad \begin{aligned} C_1(\tilde{U}) \delta_x^{up} &= \mathbb{1}_{\text{sign}(C_1 > 0)} \delta_x^- U + \mathbb{1}_{\text{sign}(C_1 < 0)} \delta_x^+ U, \\ C_2(\tilde{U}) \delta_y^{up} &= \mathbb{1}_{\text{sign}(C_2 > 0)} \delta_y^- U + \mathbb{1}_{\text{sign}(C_2 < 0)} \delta_y^+ U \end{aligned}$$

where

$$C_1(\tilde{U}) = \frac{\delta_x^+ \tilde{U} \delta_{xx} \tilde{U} + \delta_y^+ \tilde{U} \delta_x^- \delta_y^+ \tilde{U}}{|D_{\nabla}^+ \tilde{U}|_\varepsilon^3}, \quad C_2(\tilde{U}) = \frac{\delta_x^+ \tilde{U} \delta_x^+ \delta_y^- \tilde{U} + \delta_y^+ \tilde{U} \delta_{yy} \tilde{U}}{|D_{\nabla}^+ \tilde{U}|_\varepsilon^3},$$

and $\mathbb{1}_S$ is the indicator function for the set S . A numerical discussion pointing out the differences of the ADI schemes resulting from the two linearisations (4.7) and (4.11) follows in Section 5.

4.2.3 Discussion of stability restrictions for the Hundsdorfer scheme

As we are going to illustrate numerically in Section 5, a stable application of the ADI schemes to equation (1.4) depends on the choice of the regularising parameter ε . In order to use reasonably large time steps Δt , this parameter has to be taken sufficiently

large to get stable results for the numerical solution of (1.4). For the following stability consideration we use the terminology introduced in Definition 2.1 where we consider the solution continuous in space and discrete in time.

Fully explicit numerical schemes solving TV gradient flows turn out to show restrictive stability conditions related to the strength of the nonlinearity in the TV subgradient, cf. [13, 19]. On the other hand, fully implicit schemes solving (4.11) without any operator splitting are unconditionally stable. In particular, we have the following stability theorem:

Theorem 4.1. *Let u_0 be a sufficiently regular initial condition and u_n the solution of*

$$(4.16) \quad u_{n+1} = u_n - \Delta t \Delta \nabla \cdot \left(\frac{\nabla u_{n+1}}{|\nabla u_n|_\varepsilon} \right).$$

Then, the following stability estimate holds

$$(4.17) \quad \|\nabla u_{n+1}\|_\varepsilon \leq \|\nabla u_0\|_\varepsilon,$$

where $\|w\|_\varepsilon = \left(\int_\Omega (w^2 + \varepsilon) \right)^{1/2}$.

Proof. Multiplying equation (4.16) by $\Delta^{-1}(u_{n+1} - u_n)$ (where Δ^{-1} is the inverse of the negative laplacian with zero boundary conditions) and integrating over Ω we get:

$$\langle u_{n+1} - u_n, \Delta^{-1}(u_{n+1} - u_n) \rangle = \Delta t \langle \nabla \cdot \left(\frac{\nabla u_{n+1}}{|\nabla u_n|_\varepsilon} \right), u_{n+1} - u_n \rangle$$

where $\langle \cdot, \cdot \rangle$ denotes the L^2 inner product. We can rewrite the left hand side of the equation above using the properties of Δ^{-1} and applying the divergence theorem, thus finding:

$$\langle \nabla \Delta^{-1}(u_{n+1} - u_n), \nabla \Delta^{-1}(u_{n+1} - u_n) \rangle + \Delta t \left\langle \left(\frac{\nabla u_{n+1}}{|\nabla u_n|_\varepsilon} \right), \nabla(u_{n+1} - u_n) \right\rangle = 0.$$

We can now apply the result provided in [23] and summing over all $t_n = n\Delta t$ up to $T = N\Delta t$, finding the following stability estimate:

$$\|\nabla u_{n+1}\|_\varepsilon \leq \Delta t \sum_n \|\partial_t u_{n+1}\|_{-1}^2 + \|\nabla u_{N+1}\|_\varepsilon \leq \|\nabla u_0\|_\varepsilon,$$

where $\partial_t u_{n+1} = \frac{u_{n+1} - u_n}{\Delta t}$, which gives (4.17). In particular, estimate (4.17) does not depend on the size of ε . \square

These considerations about explicit and implicit schemes solving directly (i.e. without any splitting) our problem (1.4) serve as a motivation for the following estimates. We focus on the *Hundsdorfer* scheme (4.4) applied to the TV- H^{-1} equation with the choice (4.9), (4.10). In each iteration the numerical solution is computed from equations consisting of a combination of explicit and implicit quantities. In particular, the explicit quantities might affect the stability properties of the scheme. To motivate this, we focus in the following just on the first three stages of the scheme (4.4) applied to the TV- H^{-1} equation with the choice (4.9), (4.10) and $\theta = 1/2$. Considering the first three stages of (4.4) only can be justified by the fact that the subsequent three stages of the scheme have a similar structure and are not expected to change the stability properties drastically.

Combining the three steps of the scheme (4.4) with (4.9), (4.10) we find the following expression:

$$\begin{aligned}
 (4.18) \quad & \frac{u_{n+1} - u_n}{\Delta t} + \frac{1}{2} \partial_{xx}(C_1(u_n) \partial_{xx} u_{n+1}) + \frac{1}{2} \partial_{yy}(C_2(u_n) \partial_{yy} u_{n+1}) \\
 & + \frac{\Delta t}{4} \partial_{xx}(C_1(u_n) \partial_{xx} (\partial_{yy}(C_2(u_n) \partial_{yy} u_{n+1}))) \\
 & = -\Delta \nabla \cdot \left(\frac{\nabla u_n}{|\nabla u_n|_\varepsilon} \right) + \frac{1}{2} \partial_{xx}(C_1(u_n) \partial_{xx} u_n) + \frac{1}{2} \partial_{yy}(C_2(u_n) \partial_{yy} u_n) \\
 & + \frac{\Delta t}{4} \partial_{xx}(C_1(u_n) \partial_{xx} (\partial_{yy}(C_2(u_n) \partial_{yy} u_n)))
 \end{aligned}$$

where ∂_x denotes the continuous derivative with respect to x and the positive quantities $C_1(u_n)$ and $C_2(u_n)$ come from the linearisation described in Section 4.2.1. They read:

$$C_1(u_n) = \frac{\varepsilon + \partial_y(u_n)^2}{|\nabla u_n|_\varepsilon^3}, \quad C_2(u_n) = \frac{\varepsilon + \partial_x(u_n)^2}{|\nabla u_n|_\varepsilon^3}.$$

We observe that a mixed, eighth-order operator appears, both on the left and on the right hand side of (4.18). In the following stability discussion we neglect these high-order terms which only represent second-order in time contributions.

Now, we multiply equation (4.18) by u_{n+1} and integrate over the domain Ω . By applying integration by parts twice with respect to the x and the y variables to the second and the third terms of the left hand side of the equation, respectively, we get:

$$\begin{aligned}
 (4.19) \quad & \frac{1}{2} \left(\int_{\Omega} \partial_{xx}(C_1(u_n) \partial_{xx} u_{n+1}) u_{n+1} + \int_{\Omega} \partial_{yy}(C_2(u_n) \partial_{yy} u_{n+1}) u_{n+1} \right) \\
 & = \frac{1}{2} \left(\int_{\Omega} C_1(u_n) (\partial_{xx} u_{n+1})^2 + \int_{\Omega} C_2(u_n) (\partial_{yy} u_{n+1})^2 \right) \\
 & \geq \frac{1}{2} K(\varepsilon) (\|\partial_{xx}(u_{n+1})\|^2 + \|\partial_{yy}(u_{n+1})\|^2).
 \end{aligned}$$

Here $K(\varepsilon)$ is a suitable constant that depends on the regularising parameter ε only. A similar strategy is applied to the analogous terms on the right hand side, where we have also used Young's inequality with weights δ_1 and δ_2 . We obtain:

$$\begin{aligned}
 & \frac{1}{2} \left(\int_{\Omega} C_1(u_n) \partial_{xx} u_n \partial_{xx} u_{n+1} + \int_{\Omega} C_2(u_n) \partial_{yy} u_n \partial_{yy} (u_{n+1}) \right) \\
 & \leq \frac{1}{4\delta_1} \|C_1(u_n) \partial_{xx} u_{n+1}\|^2 + \frac{\delta_1}{4} \|\partial_{xx}(u_{n+1})\|^2 + \frac{1}{4\delta_2} \|C_2(u_n) \partial_{yy} u_{n+1}\|^2 + \frac{\delta_2}{4} \|\partial_{yy}(u_{n+1})\|^2.
 \end{aligned}$$

The first term on the left hand side of (4.18) can be dealt with by using Young's inequality again. It remains to consider the first, nonlinear, term on the right hand side of (4.18). By applying the divergence theorem, integration by parts and Young's inequality with weight δ_3 , we get for this term the following estimate:

$$\begin{aligned}
 (4.20) \quad & - \int_{\Omega} \Delta \nabla \cdot \left(\frac{\nabla u_n}{|\nabla u_n|_\varepsilon} \right) u_{n+1} = \int_{\Omega} \nabla \nabla \cdot \left(\frac{\nabla u_n}{|\nabla u_n|_\varepsilon} \right) \nabla u_{n+1} \\
 & = - \int_{\Omega} \nabla \cdot \left(\frac{\nabla u_n}{|\nabla u_n|_\varepsilon} \right) \Delta u_{n+1} \leq \frac{1}{2\delta_3} \left\| \nabla \cdot \left(\frac{\nabla u_n}{|\nabla u_n|_\varepsilon} \right) \right\|^2 + \frac{\delta_3}{2} \|\Delta u_{n+1}\|^2.
 \end{aligned}$$

The second term on the right hand side of the inequality above can be merged with the corresponding ones in (4.19), choosing δ_1 small enough. Denoting by $C_3(u_n) = \frac{\partial_x u_n \partial_y u_n}{|\nabla u_n|_\varepsilon^3}$, for the curvature term in (4.20) we observe that:

$$\begin{aligned} \left\| \nabla \cdot \left(\frac{\nabla u_n}{|\nabla u_n|_\varepsilon} \right) \right\|^2 &= \int_{\Omega} (C_1(u_n) \partial_{xx}(u_n) + C_2(u_n) \partial_{yy}(u_n) + C_3(u_n) \delta_{xy}(u_n))^2 \\ &\leq 2 \left(2 \left(\int_{\Omega} (C_1(u_n) \partial_{xx}(u_n))^2 + \int_{\Omega} (C_2(u_n) \partial_{yy}(u_n))^2 \right) + \int_{\Omega} (C_3(u_n) \delta_{xy}(u_n))^2 \right) \\ &\leq 4 \left(\frac{1}{\sqrt{\varepsilon}} + \frac{1}{|\partial_y(u_n)|} \right)^2 \|\partial_{xx}(u_n)\|^2 + 4 \left(\frac{1}{\sqrt{\varepsilon}} + \frac{1}{|\partial_x(u_n)|} \right)^2 \|\partial_{yy}(u_n)\|^2 + \frac{1}{\sqrt{\varepsilon}} \|\delta_{xy}(u_n)\|^2 \end{aligned}$$

where we used Cauchy's inequality and upper bounds on C_1, C_2 and C_3 . Defining $K_1(\varepsilon)$, $K_2(\varepsilon)$ and $K_3(\varepsilon)$ as

$$K_1(\varepsilon) := 4 \left(\frac{1}{\sqrt{\varepsilon}} + \frac{1}{|\partial_y(u_n)|} \right)^2, \quad K_2(\varepsilon) := 4 \left(\frac{1}{\sqrt{\varepsilon}} + \frac{1}{|\partial_x(u_n)|} \right)^2, \quad K_3(\varepsilon) := \frac{1}{\sqrt{\varepsilon}}$$

and using an estimate proved in [17] for the mixed derivative term, we get the following bound

$$\left\| \nabla \cdot \left(\frac{\nabla u_n}{|\nabla u_n|_\varepsilon} \right) \right\|^2 \leq K_1(\varepsilon) \|\partial_{xx}(u_n)\|^2 + K_2(\varepsilon) \|\partial_{yy}(u_n)\|^2 + K_3(\varepsilon) \sup_{z \in \{x, y\}} \|\delta_{zz}(u_n)\|^2.$$

Collecting the previous estimates, choosing δ_1, δ_2 and δ_3 small enough and applying once more upper bounds on C_1 and C_2 we get the following stability estimate

$$\begin{aligned} (4.21) \quad &\frac{1}{2\Delta t} \|u_{n+1}\|^2 + \|\partial_{xx}(u_{n+1})\|^2 + \|\partial_{yy}(u_{n+1})\|^2 \\ &\leq \frac{1}{2\Delta t} \|u_n\|^2 + \tilde{K}_1(\varepsilon) \|\partial_{xx}(u_n)\|^2 + \tilde{K}_2(\varepsilon) \|\partial_{yy}(u_n)\|^2 + \tilde{K}_3(\varepsilon) \sup_{z \in \{x, y\}} \|\delta_{zz}(u_n)\|^2 \end{aligned}$$

for scaled constants \tilde{K}_1, \tilde{K}_2 and \tilde{K}_3 which tend to infinity as the regularising parameter $\varepsilon \searrow 0$. For this limit the estimate then blows up, thus indicating possible unstable behaviour when choosing ε small. This will be discussed in more detail in Section 5.

4.2.4 A stable AMOS ADI scheme

In order to counteract the dependence of the stability properties of the ADI schemes (4.4) and (4.14) on the size of ε , we now consider as an alternative the AMOS operator splitting scheme (3.6)-(3.7) for solving (4.6). Due to the fully implicit character of the scheme, the hope is that stability properties improve.

To see this, both ADI methods (4.4) and (4.14) can be represented in a vectorial, multiplicative form similar to (3.5). For (4.4), the operator F_0 appearing in (3.5) is taken explicitly in time. Thus, when writing the correspondent numerical scheme in a multiplicative form, we have additional explicit terms on the right hand side. Writing (4.14) in the form (3.5) we again obtain additional explicit terms appearing due to the forward Euler steps in (4.14). This, together with the stability estimates from the previous

section, indicates possible stability restrictions for these schemes that will be confirmed by the numerical results in the following Section 5.

In the following we present an AMOS scheme for solving a slightly modified version of (1.4). In system form, this new equation reads:

$$(4.22) \quad \begin{cases} u_t = v_{xx}^1 + v_{yy}^2 \\ v^1 = -\partial_x \left(\frac{u_x}{|\nabla u|_\varepsilon} \right), \quad v^2 = -\partial_y \left(\frac{u_y}{|\nabla u|_\varepsilon} \right). \end{cases}$$

This equation is more anisotropic than the original (1.4) in the sense that the nonlinear diffusion in x and y directions are considered separately. Only the diffusion weighting involves the whole image gradient taking both x and y variations into account. In this way, linearising v_1 and v_2 by considering the diffusion weighting $1/|\nabla \tilde{u}|_\varepsilon$ for a given \tilde{u} , results in an equation with only pure x and y derivatives. Considering such an equation reduces the explicit components appearing in the scheme, allowing a fully implicit treatment of the operators, though still exploiting the advantage of directional splitting by solving along the two directions x and y separately. Applying the AMOS scheme (3.6)-(3.7) to the linearisation of (4.22), we obtain:

$$(4.23) \quad \begin{cases} \begin{pmatrix} U_* \\ V_*^2 \end{pmatrix} = \begin{pmatrix} U_n + \Delta t F_2(U_*, V_*^2) \\ G_2(U_*, V_*^2) \end{pmatrix}, \\ \begin{pmatrix} \tilde{U}_{n+1} \\ \tilde{V}_{n+1}^1 \end{pmatrix} = \begin{pmatrix} U_* + \Delta t F_1(\tilde{U}_{n+1}, \tilde{V}_{n+1}^1) \\ G_1(\tilde{U}_{n+1}, \tilde{V}_{n+1}^1) \end{pmatrix} \\ \begin{pmatrix} U_* \\ V_*^1 \end{pmatrix} = \begin{pmatrix} U_n + \Delta t F_1(U_*, V_*^1) \\ G_1(U_*, V_*^1) \end{pmatrix}, \\ \begin{pmatrix} \bar{U}_{n+1} \\ \bar{V}_{n+1}^2 \end{pmatrix} = \begin{pmatrix} U_* + \Delta t F_2(\bar{U}_{n+1}, \bar{V}_{n+1}^2) \\ G_2(\bar{U}_{n+1}, \bar{V}_{n+1}^2) \end{pmatrix}, \\ \begin{pmatrix} U_{n+1} \\ V_{n+1}^1 \\ V_{n+1}^2 \end{pmatrix} = \begin{pmatrix} \frac{\tilde{U}_{n+1} + \bar{U}_{n+1}}{2} \\ \frac{\tilde{V}_{n+1}^1 + V_*^1}{2} \\ \frac{\bar{V}_{n+1}^2 + V_*^2}{2} \end{pmatrix} \end{cases}$$

where the operators are defined exactly as in (4.13) and upwinding is used for the first derivatives as in (4.15). We recall that the alternating application of the scheme first in the $y - x$ direction and subsequently in the $x - y$ direction allows to achieve order two of accuracy, as explained in [4]. As we will see in Section 5, the scheme (4.23) has better stability properties where the choice of the time step does not seem to depend on the size of ε , thus suggesting – at least empirically – unconditional stability.

4.3 Primal-dual formulation of TV- H^{-1} equation with penalty term

An interesting alternative to the linearisations of the primal formulation of equation (1.4) is motivated by earlier work on numerically characterising elements in the subdifferential of the total variation seminorm by primal-dual iterations, see [5, 39, 6] for instance. In

what follows we briefly outline such a strategy combined with ADI splitting for solving equation (1.4), that is

$$(4.24) \quad u_t = \Delta q \quad \text{in } \Omega \times (0, \infty), \quad q \in \partial|Du|(\Omega).$$

Here, $q \in \partial|Du|(\Omega)$ by definition of the subdifferential means

$$(4.25) \quad q \in \partial|Du|(\Omega) \iff |Du|(\Omega) - \int_{\Omega} qu \, dx \leq |Dv|(\Omega) - \int_{\Omega} qv \, dx, \quad \forall v \in L^2(\Omega).$$

Equivalently, if $u \in BV(\Omega) \subset L^2(\Omega)$ solves the variational problem

$$(4.26) \quad \min_{v \in BV(\Omega)} \left\{ |Dv|(\Omega) - \int_{\Omega} qv \, dx \right\},$$

then, by definition of being a minimum, u fulfills (4.25), that is $q \in \partial|Du|(\Omega)$. Inserting the definition of the total variation seminorm (1.2) into (4.26) we receive

$$(4.27) \quad \min_{v \in BV(\Omega)} \left\{ \sup_{\mathbf{p} \in C_0^\infty(\Omega; \mathbb{R}^2), \|\mathbf{p}\|_\infty \leq 1} \int_{\Omega} \nabla \cdot \mathbf{p} v \, dx - \int_{\Omega} qv \, dx \right\}$$

which is typically known as the *primal-dual* formulation of the problem (4.24). The constraint on \mathbf{p} appearing in (4.27) can be relaxed, for instance, by a penalty method. That is, we remove the constraint from the minimisation in (4.27) and instead add a term that penalises the functional if $\|\mathbf{p}\|_\infty > 1$. A typical example for such a penalty term F is

$$F(s) = \frac{1}{2} \|\max\{s, 0\}\|_2^2.$$

With these considerations we reformulate (4.27) as

$$(4.28) \quad \min_{v \in BV(\Omega)} \sup_{\mathbf{p} \in C_0^\infty(\Omega; \mathbb{R}^2)} \left\{ \int_{\Omega} \nabla \cdot \mathbf{p} v \, dx - \frac{1}{\varepsilon} F(|\mathbf{p}| - 1) - \int_{\Omega} qv \, dx \right\}$$

where the parameter $1 \gg \varepsilon > 0$ is the weight of our penalisation. We can then find the optimality conditions for solutions \mathbf{p} and u of (4.28) which, merged with equation (4.24), allow us to consider the following, approximate formulation of (1.4)

$$(4.29) \quad \begin{cases} u_t = \Delta q, \\ q = \nabla \cdot \mathbf{p}, \\ 0 = -\nabla u - \frac{1}{\varepsilon} H(\mathbf{p}). \end{cases}$$

In the system above we have indicated by H the derivative of the penalty term $F(|\mathbf{p}| - 1)$, i.e.

$$H(\mathbf{p}) = \mathbb{1}_{\{|\mathbf{p}| \geq 1\}} \text{sgn}(\mathbf{p})(|\mathbf{p}| - 1),$$

which we linearise via its first-order Taylor approximation, that is

$$H(\mathbf{p}) \approx H(\tilde{\mathbf{p}}) + H'(\tilde{\mathbf{p}})(\mathbf{p} - \tilde{\mathbf{p}}).$$

Here, H' denotes the Jacobian of H . In order to guarantee the invertibility of the now linear operator that defines the system (4.29) we add an additional damping term in \mathbf{p} as suggested, for instance, in [39]. Collecting everything, we propose the following numerical scheme for solving (4.29),

$$(4.30) \quad \left\{ \begin{array}{l} \frac{U_{n+1}^{(k)} - U_n}{\Delta t} = D_{\Delta} Q_{n+1}^{(k)}, \\ Q_{n+1}^{(k)} = D_{\text{div}}^{-} \mathbf{P}_{n+1}^{(k)}, \\ 0 = -D_{\nabla}^{+} U_{n+1}^{(k)} - \frac{1}{\varepsilon} H(\mathbf{P}_{n+1}^{(k-1)}) \\ \quad - \frac{1}{\varepsilon} H'(\mathbf{P}_{n+1}^{(k-1)})(\mathbf{P}_{n+1}^{(k)} - \mathbf{P}_{n+1}^{(k-1)}) - \tau^k (\mathbf{P}_{n+1}^{(k)} - \mathbf{P}_{n+1}^{(k-1)}), \end{array} \right.$$

which consists of two nested iterations, the inner damped Newton iteration with index k and an outer time stepping with index n for the evolution in time of U . Here, τ^k is a sequence of parameters controlling the damping of the Newton iteration in every time step. The authors in [39, 6] suggest to start with a large value τ^0 and then decrease it in the inner iterations to ensure efficient convergence.

System (4.30) could now be discretised in space and either solved directly using Newton's iterations (compare [6] where the authors used this strategy to solve equation (1.6)) or fitted into a numerical ADI scheme. A detailed presentation and analysis of the resulting scheme is beyond the scope of the present paper and a matter of future research.

5 Numerical results

In this section we discuss the numerical performance of the ADI methods proposed in this paper. To do so, we present numerical experiments for a Gaussian, an oscillatory function made up of sine and cosine functions and grayscale images as initial conditions. The paper is furnished with numerical examples for image inpainting.

We start presenting the numerical results obtained applying the *Hundsdorfer* scheme (3.4) to compute the numerical solution of the biharmonic equation (4.2) and to the equivalent system (4.3). Next, we report on numerical experiments for the TV- H^{-1} equation (1.4) using the *Hundsdorfer* scheme (4.4) with (4.9)-(4.10). The choice of the time step size Δt of this scheme is constrained by very strong stability restrictions related to the size of the regularising parameter ε . A possible dependence of this type for stability was already discussed in Section 4. Our numerical tests for the application of the *Peaceman-Rachford* scheme (4.14) to the TV- H^{-1} equation show the same stability behaviour and are therefore not included in the paper. Finally, we show the numerical results for the solution of the modified TV- H^{-1} system (4.22) solved with the AMOS scheme (4.23). This scheme shows stable behaviour independent of the size of Δt and ε .

5.1 The biharmonic equation: numerical results

The linear system that arises from the application of the *Hundsdorfer* scheme (4.4) with (4.5) to the biharmonic equation (4.3) is solved by the Schur complement method.

We consider a grid of 100×100 grid points for the discretisation of the spatial domain Ω being the unit square. We analyse the example of the evolution of the biharmonic equation having as initial condition U_0 the Gaussian density $U_{ij}^0 = \exp(-((x_i - 1/2)^2 + (y_j - 1/2)^2)/\gamma^2)$ where the variance γ^2 is equal to 100. Figure 1 shows two iterates of the scheme with $\theta = \sigma = 1/2$ for a time-step size $\Delta t = C(\Delta x)^2$, where C now and for the rest of the discussion is equal to 0.1. Even for such a big choice of Δt the result is stable and the convergence of the solution to the steady state is quick. Considering another initial condition U_0 and taking, for instance, some very oscillatory function does not effect the performance of the method. This confirms the unconditional stability of this scheme when applied to a linear fourth-order equation such as the biharmonic equation (4.2), compare [28]. In the following section we will see that the unconditional stability of the *Hundsdorfer* scheme breaks down when applied to the nonlinear fourth-order diffusion equation TV- H^{-1} (1.4).

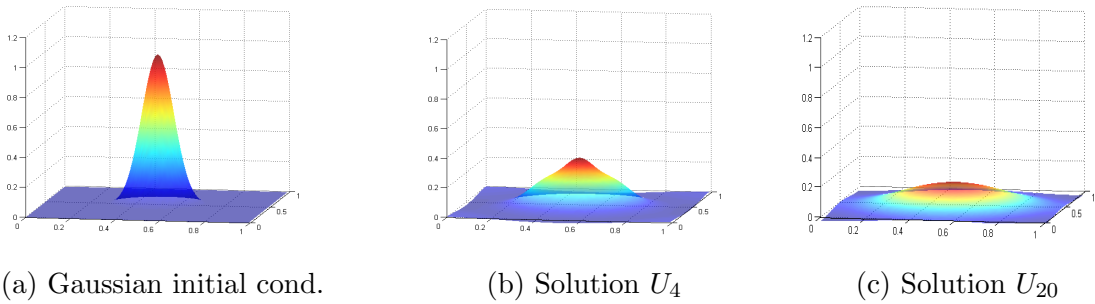


Figure 1: Evolution of the biharmonic equation (4.3) solved with the *Hundsdorfer* scheme (4.4) with $\Delta t = C(\Delta x)^2$.

5.2 TV- H^{-1} equation: numerical results

In what follows we provide numerical discussion for applying the directional splitting schemes introduced in Section 4 for the numerical solution of the regularised TV- H^{-1} equation (4.6). In particular we consider *Hundsdorfer* ADI scheme (4.4) with (4.9), (4.10) and the AMOS scheme (4.23). Once again we use the Schur complement technique to solve the linear systems that arise in the numerical solution of these schemes. The expected edge-preserving behaviour of the TV- H^{-1} equation (1.4) which is due to the subgradient of the TV functional is closely related to the size of the regularising parameter ε used in (4.6). This parameter, in fact, becomes a “measure” of how close the nonlinear diffusion is to the linear, biharmonic one. Choosing ε too big the smoothing behaviour of (4.6) is similar to the one of the biharmonic equation (4.2). In this case the stability properties of the *Hundsdorfer* scheme applied to the nonlinear equation are close to the ones discussed for the biharmonic equation in the previous section. On the other hand, small values of ε keep the regularised version of the subgradient of the total variation close to its exact characterisation and hence solution show edge-preserving features. However, stability issues that disturb and worsen the performance of the methods appear. The explicit treatment of some of the terms in the *Hundsdorfer* and the *Peaceman-Rachford* schemes (namely, the mixed derivative term in (4.4) and the half-direction forward Euler steps in (4.14)) seem to influence the stability of the schemes in a negative way. In particular, the

time step sizes Δt have to be decreased significantly with small values of ε . Moreover, the choice of the initial condition also influences the stability properties. For instance, for smooth Gaussian initial conditions with large support the time steps can be chosen larger than for an oscillatory initial condition, see Figures 2-3. These issues are resolved by the AMOS scheme (4.23) which did not show dependence of Δt on the size of ε nor on the type of initial condition in order to get stable results.

In the following we present the numerical results obtained considering the linearisation of the system (4.6) given by (4.7) and solved by the *Hundsdorfer* ADI method (4.4). We consider as initial conditions both the Gaussian density $U_{ij}^0 = \exp(-((x_i - 1/2)^2 + (y_j + 1/2)^2)/\gamma^2)$ with $\gamma^2 = 100$ and the oscillatory function $U_{ij}^0 = \sin(8\pi x_i) + \cos(8\pi y_j)$. Figure 2 shows iterates of the *Hundsdorfer* scheme with $\theta = \sigma = 1/2$, $\varepsilon = 5$ and $\Delta t = C(\Delta x)^3$ applied to the Gaussian datum. The value $\varepsilon = 5$ is the smallest possible value that can be used in order to get stable solutions of the *Hundsdorfer* scheme with $\Delta t = C(\Delta x)^3$. Figure 3 shows the evolution of the process for the oscillatory datum with the same choice of ε as before. In this case to get stable results smaller time steps are needed, thus showing stability dependence on the initial condition.

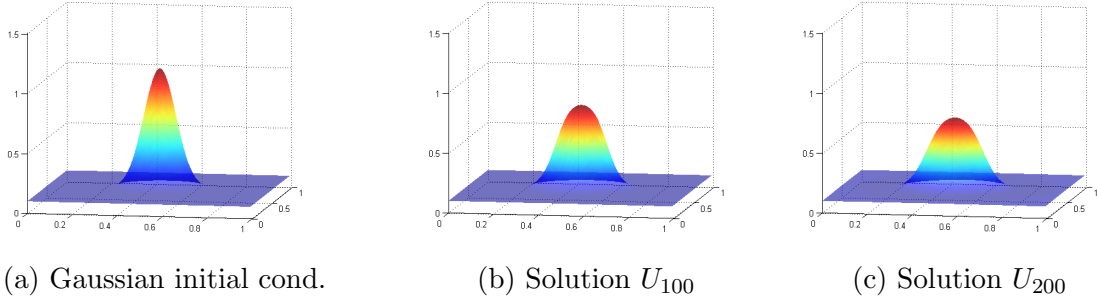


Figure 2: Evolution of the TV- H^{-1} equation (1.4) by the *Hundsdorfer* scheme (4.4) with $\Delta t = C(\Delta x)^3$ and $\varepsilon = 5$.

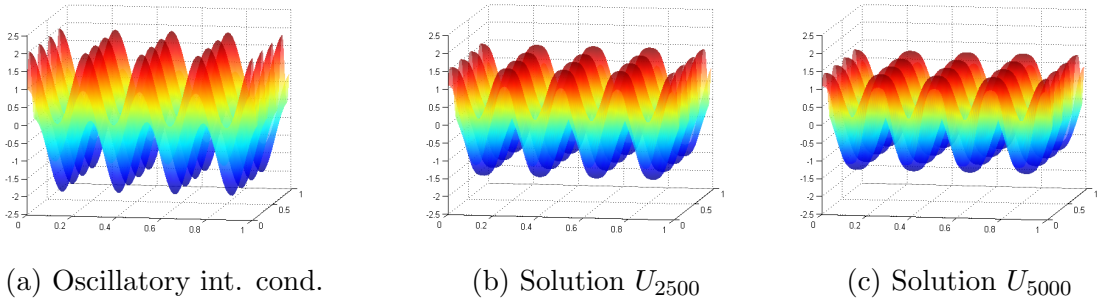


Figure 3: Evolution of the TV- H^{-1} equation (1.4) by the *Hundsdorfer* scheme (4.4) with decreased $\Delta t = C(\Delta x)^4$ and $\varepsilon = 5$ to control stability.

In Figures 4–5 we describe this stability dependence in more detail. For the two different choices of the initial conditions considered above, the smallest values ε needed for stability as the size of Δt increases are plotted. To compare the different setups of the *Hundsdorfer* scheme with respect to θ , these tests were performed for different values of the stabilising parameter $\theta = 0, 1/2, 1/2 + \sqrt{3}/6, 1$ thus resulting into four graphs per test. Note that different values of θ (compare [31, 32, 28, 29] for such choices) result into

different weighting of implicit and explicit terms, that means considering a fully explicit method for $\theta = 0$ and implicit contributions in the unidirectional steps of (4.4) in the other cases (see Section 3.2). For each of these graphs, their epigraph corresponds to the region of *stability* of the method (according to Definition 2.1). To obtain such graphs we have considered different values of the constant C and of the regularising parameter ε for time step sizes of the order $(\Delta x)^k, k = 2, 3, 4$. For the choice $\theta = 0$ stable solutions are only obtained by very restrictive choices of ε . This situation improves for $\theta > 0$. In particular, for values of θ close to 1 the stability constraint on the size of ε is reduced. However, in all cases, these plots show a clear dependence of the stability of the *Hundsdorfer* scheme on the strength of the nonlinearity (encoded by the size of ε). This creates numerical difficulties in the attempt of increasing the time step size to get an efficient numerical scheme for solving the approximated problem (4.6) for sufficiently small values of ε .

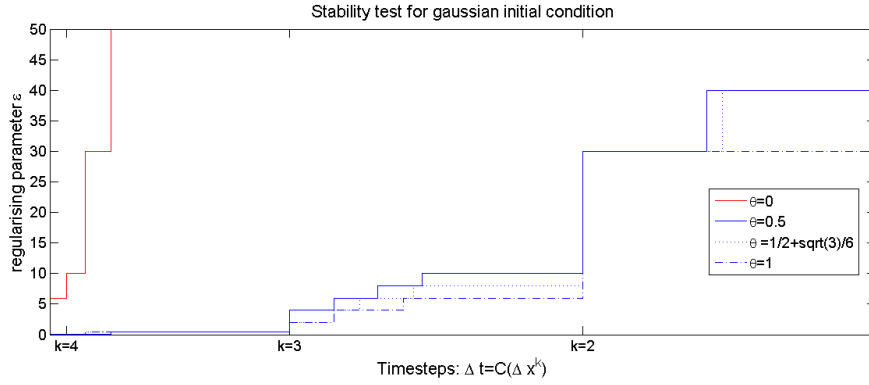


Figure 4: Stability test for the numerical solution of (4.7) solved with the *Hundsdorfer* scheme (4.4) with initial condition $U_{ij}^0 = \exp -(((x_i - 1/2)^2 + (y_j + 1/2)^2)/\gamma^2)$ with $\gamma^2 = 100$ for different choices of stabilising parameter θ . For each time step size the minimum value of ε for which we get stability is plotted.

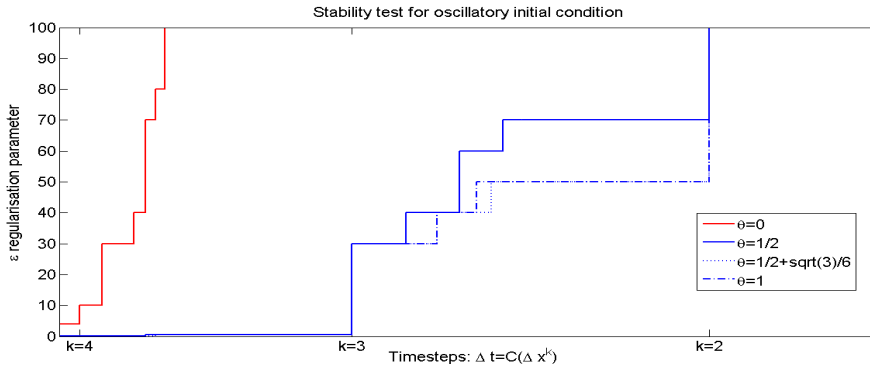


Figure 5: Stability test for the numerical solution of (4.7) solved with the *Hundsdorfer* scheme (4.4) with initial condition $U_{ij}^0 = \sin(8\pi x_i) + \cos(8\pi y_j)$. Comparison with Figure 4 shows dependence on the initial condition for admissible values of ε providing stability.

We do not present here the numerics related to the application of the *Peaceman-Rachford* method (4.14) to the TV- H^{-1} equation (4.22) as the stability issues resemble the ones described above. In order to overcome such problems, we present in the following

the results related to the application of the ADI AMOS scheme (4.23) solving the slightly modified system (4.22).

Due to the implicit character of the scheme (4.23), improved stability properties are expected while keeping the advantages of the directional splitting strategy, that is each can be solved very efficiently. In the Figures 7-8 we show the evolution of the TV- H^{-1} equation (4.22) for the Gaussian initial condition as above for $\Delta t = C(\Delta x)^3$ and $\Delta t = C(\Delta x)^2$ with fixed $\varepsilon = 0.001$. We observe that the time discretisation provides stable results even for large Δt . However, as clearly visible in Figure 8, choosing Δt too large badly affects time accuracy.

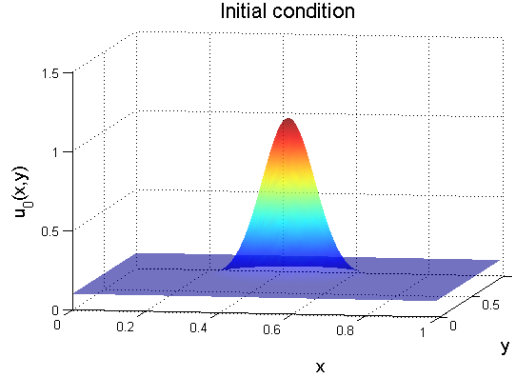


Figure 6: Initial condition

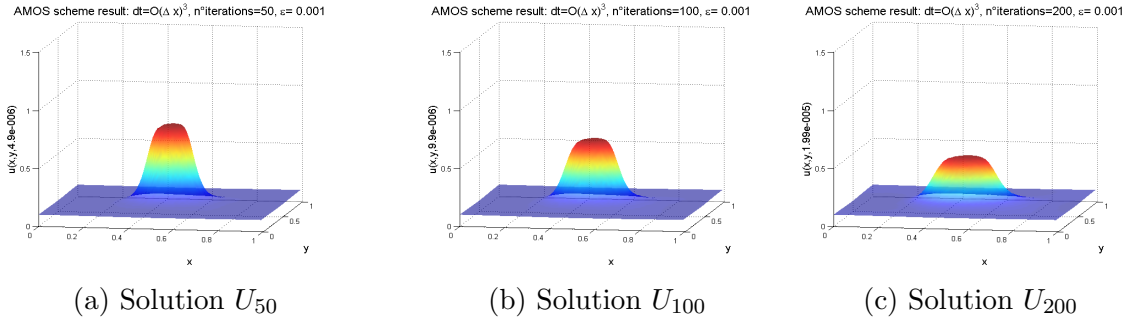


Figure 7: Evolution of the modified TV- H^{-1} equation (4.22) with the AMOS scheme (4.23) with $\Delta t = C(\Delta x)^3$ and $\varepsilon = 0.001$.

The convergence rate of the AMOS scheme (4.23) for the choice of $\Delta t = C(\Delta x)^3$ and regularising parameter $\varepsilon = 0.001$ when applied to the Gaussian initial condition is presented in Figure 9. We observe an exponential-type decay for the ℓ^∞ norm of the difference between the iterative solution U_n of (4.23) and the steady state U_∞ which has been computed numerically beforehand by iterating the scheme till the quantity $\|U_{n+1} - U_n\|_\infty / (MN) \leq 10^{-10}$. For comparison, the exponential function $(7 \cdot 10^{-6})e^{-0.05x}$, has been plotted. Figure 10 shows the decay of total variation energy (1.2) towards the energy of the steady state U_∞ against the number of iterations. This shows that although the modified TV- H^{-1} equation (4.22) is not as the same as the gradient flow of the total variation in the space H^{-1} , the total variation energy is still decreasing in every iteration.

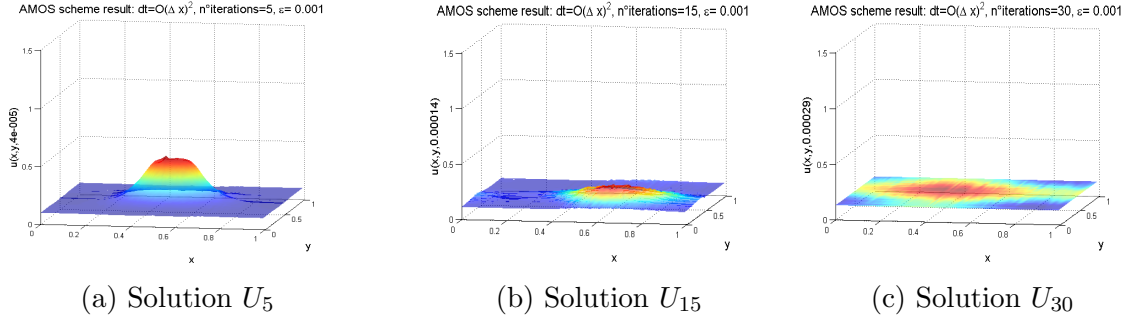


Figure 8: Evolution of the modified $\text{TV-}H^{-1}$ equation (4.22) with the AMOS scheme (4.23) with $\Delta t = C(\Delta x)^2$ and $\varepsilon = 0.001$.

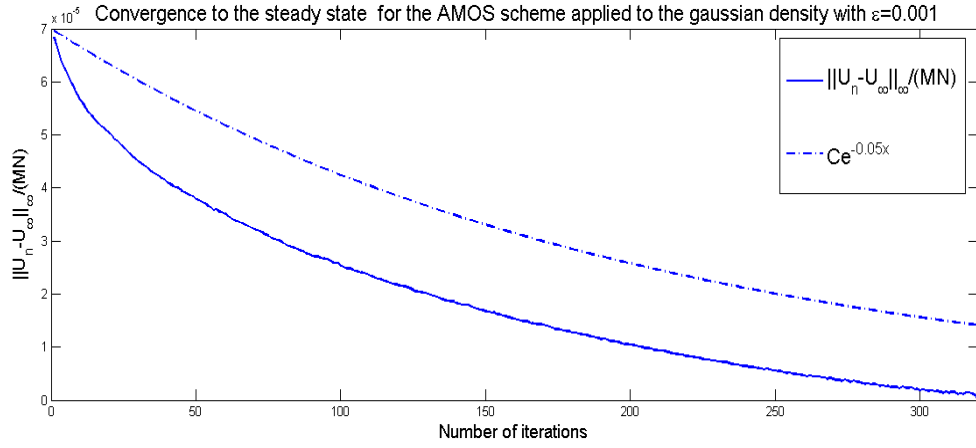


Figure 9: Convergence to the steady state for the numerical solution of the modified $\text{TV-}H^{-1}$ equation (4.22) computed with the AMOS scheme (4.23).

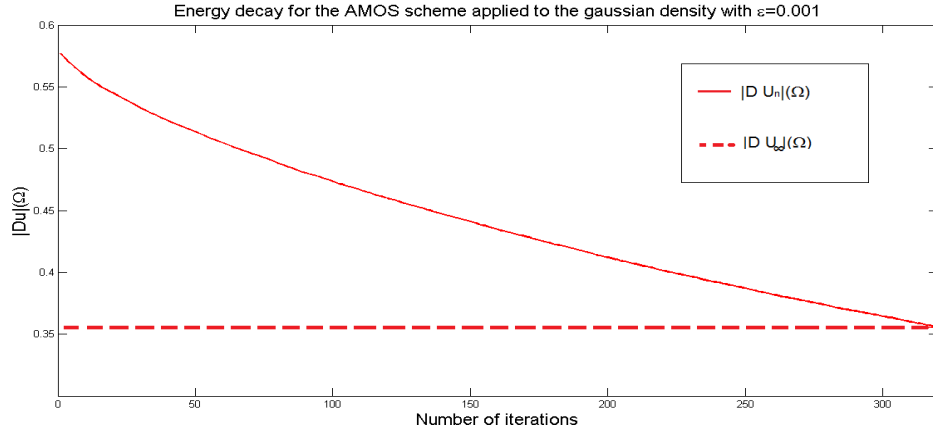


Figure 10: Energy decay for the numerical solution of the modified $\text{TV-}H^{-1}$ equation (4.22) computed with the AMOS scheme (4.23).

Motivated by our original purposes of applying the ADI schemes in the imaging framework, we present in Figure 11 the scale space properties of system (4.22) solved with the AMOS scheme (4.23) for a 120×120 image. The time step size is $\Delta t = C(\Delta x)^3$ and the regularising parameter $\varepsilon = 0.001$. Due the nonlinear nature of the equation, the diffusion

is anisotropic.

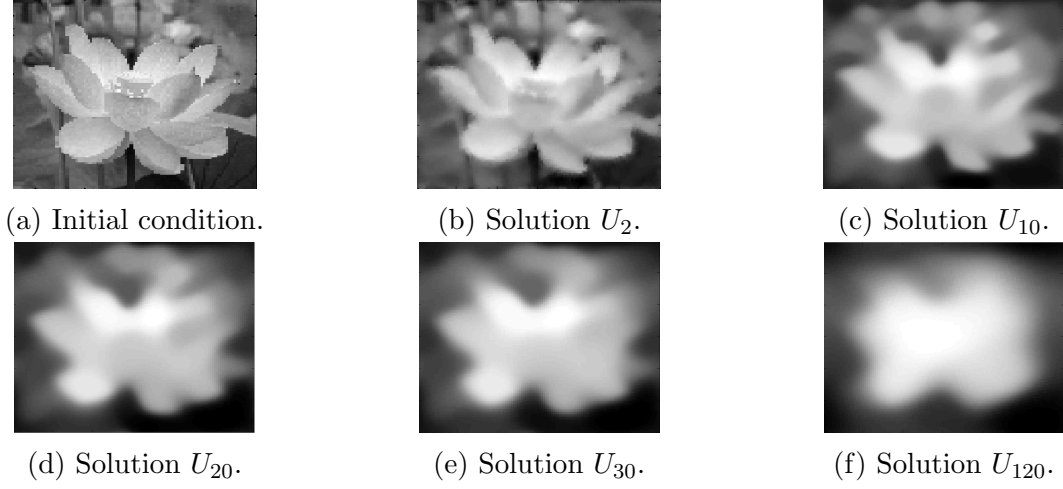


Figure 11: Scale space properties for the iterates of the numerical solution of (4.22) computed with the AMOS scheme (4.23), $\Delta t = C(\Delta x)^3$, $\varepsilon = 0.001$.

Finally, we present some numerical results obtained by using the AMOS scheme (4.23) to solve the H^{-1} -inpainting model (1.5) with the modified TV- H^{-1} equation given by (4.22). The constraint $u = f$ outside of the inpainting domain is approximately enforced by adding the fidelity term $\lambda \cdot \mathbb{1}_{\Omega \setminus D}(f - u)$ to the equation, where λ measures how close the reconstructed image is to the original one. In Figure 12 we show the result for inpainting a 150×150 image of a cross: the initial condition is inpainted in 1000 iterations using a time step size of $\Delta t = C(\Delta x)^3$ and the regularising parameter ε is chosen to be $\varepsilon = 0.001$, thus allowing the preservation of edges additionally to fulfilling the connectivity principle that is a consequence of the fourth-order of the method.

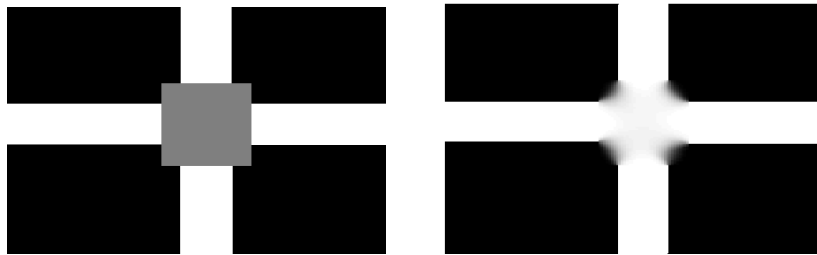


Figure 12: Solution of inpainting problem obtained with 1000 iterations of the ADI AMOS scheme (4.23), $\Delta t = C(\Delta x)^3$, $\varepsilon = 0.001$.

As a second example, we consider a grayvalue 300×300 photograph of a toucan in Figure 13. Its reconstruction is obtained in only 20 iterations using the AMOS scheme (4.23) with $\Delta t = C(\Delta x)^3$ and $\varepsilon = 0.001$.

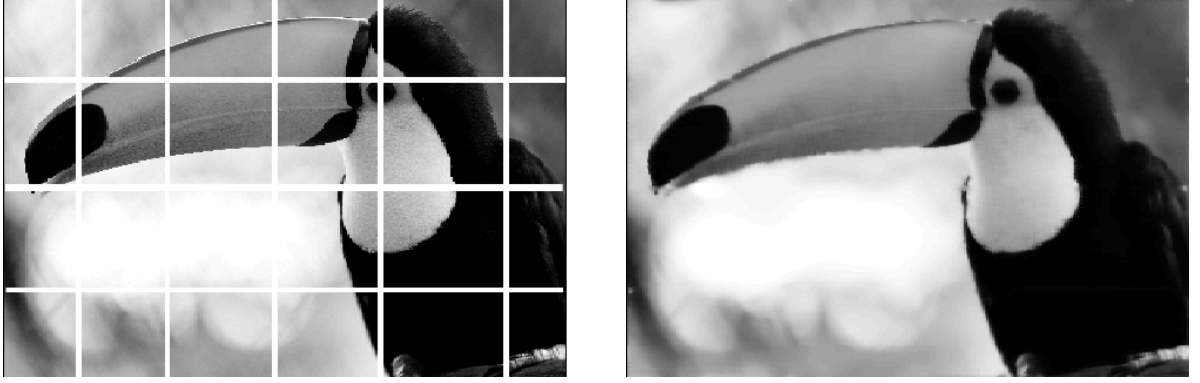


Figure 13: Solution of the inpainting problem obtained with ADI AMOS scheme (4.23) after 20 iterations, $\Delta t = C(\Delta x)^3$, $\varepsilon = 0.001$.

6 Conclusion

In this paper we have studied the applicability of directional splitting techniques to fourth-order nonlinear diffusion equations. In particular we have considered the $\text{TV-}H^{-1}$ equation which is the H^{-1} gradient flow of the total variation that emerges from imaging applications such image denoising and inpainting.

The numerical challenges when solving this equation are both its fourth-differential order and the strong nonlinearity due to the total variation subgradient in the equation. In former work on numerical schemes for this equation these issues have been either addressed by using explicit time stepping schemes with tiny step sizes, semi-implicit schemes with heavy damping, e.g. [48], or by fully implicit schemes, e.g. [19], which are computationally expensive to solve. Having this in mind directional splitting seems to be a promising compromise. The computational cost of each of its iterations is very small due to the triangular form of the system matrices, while the stability conditions are improved when compared to an explicit in time discretisation.

We have proposed three different ADI methods applied to two linearisations of the $\text{TV-}H^{-1}$ equation and presented numerical results for Gaussian-type initial conditions, image smoothing and image inpainting. Our investigation of stability properties of the schemes results into the following observation: any explicit terms, due to the type of splitting used, result into heavy stability conditions on the size of the time steps. For fourth-order equations these restrictions usually turn out to make a choice of Δt as small as $(\Delta x)^4$ necessary, thus preventing an efficient numerical solution of the equation. In particular, even if each iteration of the numerical scheme is very cheap – due to its “one-dimensional” character – a large number of them have to be computed in order to see any significant change of the solution in time. The AMOS scheme provides a stable and cheap numerical scheme for solving a slightly modified version of the $\text{TV-}H^{-1}$ equation, which turns this fourth-order nonlinear equation numerically tractable and hence makes it more attractive for imaging applications.

Acknowledgements. Carola-Bibiane Schönlieb acknowledges the financial support provided by the Cambridge Centre for Analysis (CCA), the Royal Society International

Exchanges Award IE110314 for the project *High-order Compressed Sensing for Medical Imaging*, the EPSRC first grant Nr. EP/J009539/1 *Sparse & Higher-order Image Restoration*, and the EPSRC / Isaac Newton Trust Small Grant on *Non-smooth geometric reconstruction for high resolution MRI imaging of fluid transport in bed reactors*. Further, this publication is based on work supported by Award No. KUK-I1-007-43 , made by King Abdullah University of Science and Technology (KAUST).

References

- [1] L. Ambrosio, N. Fusco, D. Pallara, *Functions of Bounded Variation and Free Discontinuity Problems*, Oxford Mathematical Monographs, Oxford, (2000).
- [2] J.-F. Aujol, A. Chambolle, *Dual norms and image decomposition models*, International Journal of Computer Vision, volume 63, number 1, pages 85-104, June 2005.
- [3] J.-F. Aujol, G. Gilboa, *Constrained and SNR-based Solutions for TV-Hilbert Space Image Denoising*, Journal of Mathematical Imaging and Vision, volume 26, numbers 1-2, pages 217-237, November 2006.
- [4] D. Barash, M. Israeli, R. Kimmel, *An accurate operator splitting scheme for nonlinear diffusion filtering*, In: Scale-Space and Morphology in Computer Vision, M. Kerckhove (ed.), Lecture Notes in Computer Science 2106, 281-289, (2006).
- [5] M. Benning, *Singular regularization of inverse problems*, PhD thesis, University of Münster, (2011).
- [6] M. Benning, L. Calatroni, B. Düring, C.-B. Schönlieb, *A primal-dual approach for a total variation Wasserstein flow*, to appear in GSI 2013 LNCS proceedings, Springer, (2013).
- [7] M. Burger, M. Franek, C.-B. Schönlieb, *Regularised Regression and Density estimation based on Optimal Transport*, Appl. Math. Res. Express (AMRX) 2012 (2), 209-253, (2012).
- [8] M. Burger, L. He, C.-B. Schönlieb, *Cahn-Hilliard inpainting and a generalization for grayvalue images*, SIAM J. Imaging Sci., 2 (4), 1129-1167, (2009).
- [9] T. F. Chan, and J. J. Shen, *Image Processing and Analysis Variational, PDE, wavelet, and stochastic methods*, SIAM, (2005).
- [10] A. Chambolle, *An algorithm for total variation minimization and applications*, J. Math. Imaging Vis., 20 (1-2), 89-97, (2004).
- [11] A. Chambolle, P.-L. Lions, *Image recovery via total variation minimization and related problems.*, Numer. Math. **76**(2), 167-188, (1997).
- [12] T.F. Chan, S.H. Kang, J. Shen, *Euler's elastica and curvature-based inpainting*, SIAM J. Appl. Math., Vol. 63, Nr.2, 564-592, (2002).
- [13] T.F. Chan, P. Mulet, *On the Convergence of the Lagged Diffusivity Fixed Point Method in Total Variation Image Restoration*, SIAM Journal on Numerical Analysis, Vol. 36, No. 2., 354-367, (1999).
- [14] T.F. Chan, J. Shen, *Mathematical models for local non-texture inpaintings*, SIAM J. Appl. Math., 62(3), 1019-1043, (2001).
- [15] T.F. Chan, J. Shen, *Non-texture inpainting by curvature driven diffusions (CDD)*, J. Visual Comm. Image, Rep. 12 (4), 436-449, (2001).
- [16] T.F. Chan, J. Shen, *Variational Image Inpainting*, Comm. Pure Applied Math, Vol. 58, 579-619, (2005).
- [17] W. Chen, Z. Ditzian, *Mixed and Directional Derivatives*, Proceedings of the American Mathematical Society, 108 (1), 177-185, (1990).
- [18] S.D. Conte, R. T. James, *An alternating direction method for solving the biharmonic equation*, Math. Tables Aids Comput., 12, 198-205, (1958).
- [19] S. Didas, J. Weickert, B. Burgeth, *Stability and local feature enhancement of higher order nonlinear diffusion filtering*, in W. Kropatsch, R. Sablatnig, A. Hanbury (Eds.): Pattern Recognition. Lecture Notes in Computer Science, Vol. 3663, 451-458, Springer, Berlin, 2005.

- [20] B. Düring, C.-B. Schönlieb, *A high-contrast fourth-order PDE from imaging: numerical solution by ADI splitting*, In: *Multi-scale and High-Contrast Partial Differential Equations*, H. Ammari et al. (eds.), pp. 93-103, Contemporary Mathematics 577, American Mathematical Society, Providence, (2012).
- [21] B. Düring, D. Matthes, J.-P. Milisic, *A gradient flow scheme for nonlinear fourth order equations*, Discrete Contin. Dyn. Syst. Ser. B, 14 (3), 935-959, (2010).
- [22] C.M. Elliott, D. A. French, *Numerical studies of the Cahn-Hilliard equation for phase separation*, IMA J. Appl. Math., 38 (2), 97-128, (1987).
- [23] C.M. Elliott, S. A. Smitheman, *Analysis of the TV regularization and H^{-1} fidelity model for decomposing an image into cartoon plus texture*, Commun. on Pure and Appl. Anal., 6 (4), 917936, (2007).
- [24] C.M. Elliott, S. A. Smitheman, *Numerical analysis of the TV regularization and H^{-1} fidelity model for decomposing an image into cartoon plus texture*, IMA Journal of Numerical Analysis, 139, (2008).
- [25] S. Esedoglu, J.-H. Shen, *Digital inpainting based on the Mumford-Shah-Euler image model*, Eur. J. Appl. Math., 13:4, 353-370, (2002).
- [26] T. Goldstein, S. Osher *The split Bregman method for L^1 regularized problems*, SIAM Journal on Imaging Sciences, 2 (2), 323-343, (2009)
- [27] L. Gosse, G. Toscani, *Lagrangian numerical approximations to one-dimensional convolution-diffusion equations*, SIAM J. Sci. Comput., 28 (4), 1203-1227, (2006)
- [28] K.J. in 't Hout, B.D. Welfert, *Stability of ADI schemes applied to convection-diffusion equations with mixed derivative terms*, Appl. Numer. Math., 57, 19-35, (2007).
- [29] K.J. in 't Hout, B.D. Welfert, *Unconditional stability of second-order ADI schemes applied to multi-dimensional diffusion equations with mixed derivative terms*, Appl. Numer. Math., 59, 677-692, (2009).
- [30] P.J. van der Houwen, J.G. Verwer, *One-step splitting methods for semi-discrete parabolic equations*, Computing, 22, 291-309, (1979).
- [31] W. Hundsdorfer, J.G. Verwer, *Numerical solution of time-dependent advection-diffusion-reaction equations*, Berlin: Springer-Verlag, (2003).
- [32] W. Hundsdorfer, *Accuracy and stability of splitting with Stabilizing Corrections*, Appl. Numer. Math., 42, 213-233, (2002).
- [33] L. Lieu, L. Vese, *Image restoration and decomposition via bounded total variation and negative Hilbert-Sobolev spaces*, Applied Mathematics & Optimization, Vol. 58, pp. 167-193, 2008.
- [34] T. Lu, P. Neittaanmäki, X.-C. Tai, *A parallel splitting up method and its application to Navier-Stokes equations*, Appl. Math. Lett. 4 (2), pp. 25-29, (1991).
- [35] T. Lu, P. Neittaanmäki, X.-C. Tai, *A parallel splitting-up method for partial differential equations and its applications to Navier-Stokes equations*, RAIRO Modél. Math. Anal. Numér. 26 (6), pp. 673-708, (1992).
- [36] O.M. Lysaker, X.-C. Tai, *Iterative image restoration combining total variation minimization and a second-order functional*, International Journal of Computer Vision, Vol.66, Nr.1, pp.5-18, 2006.
- [37] S. Masnou, J. Morel, *Level Lines based Disocclusion*, 5th IEEE Int'l Conf. on Image Processing, Chicago, IL, Oct. 4-7, 1998, pp.259-263, 1998.
- [38] Y. Meyer, *Oscillating Patterns in Image Processing and Nonlinear Evolution Equations*, Univ. Lecture Ser. 22, AMS, Providence, RI, 2002.
- [39] J. Müller, *Parallel total variation minimization*, Diploma Thesis, University of Münster, WWU, (2008).
- [40] D. Mumford, B. Gidas, *Stochastic models for generic images*, Quart. Appl. Math., 59, pp. 85-111, (2001).
- [41] T.G. Myers, *Thin films with high surface tension*, SIAM Rev., 40 (3), 441-462, (1998).
- [42] A. Novick-Cohen, L. A. Segel, *Nonlinear aspects of the Cahn-Hilliard equation*, Phys. D, 10 (3), 277-298, (1984).
- [43] A. Oron, S.H. Davis, S.G. Bankoff, *Long-scale evolution of thin liquid films*, Phys. Rev. Lett., 85 (10), 2108-2111, (2000).
- [44] S. Osher, A. Sole, L. Vese, *Image decomposition and restoration using total variation minimization and the H^{-1} norm*, Multiscale Modelling and Simulation: A SIAM Interdisciplinary Journal, 1 (3), 349-370, (2003).

- [45] D.W. Peaceman, H.H. Rachford Jr., *The numerical solution of parabolic and elliptic differential equations*, J. Soc. Industr. Appl. Math, 3 (1), 28-41, (1955).
- [46] L.I. Rudin, S. Osher, E. Fatemi, *Nonlinear total variation based noise removal algorithms*, Physica D 60, pp. 259-268, 1992.
- [47] L. Rudin, S. Osher, *Total variation based image restoration with free local constraints*, Proc. 1st IEEE ICIP, 1, 31-35, (1994).
- [48] C.-B. Schönlieb, A. Bertozzi, *Unconditionally stable schemes for higher order inpainting*, Commun. Math. Sci., 9 (2), 413-457, (2011).
- [49] C.-B. Schönlieb, A. Bertozzi, M. Burger, L. He, *Image inpainting using a fourth-order total variation flow*, Proc. Int. Conf. SampTA09, Marseilles, (2009).
- [50] C.-B. Schönlieb, *Total variation minimization with an H^{-1} constraint*, CRM Series 9, Singularities in Nonlinear Evolution Phenomena and Applications Proceedings, Scuola Normale Superiore Pisa, 201232, (2009).
- [51] P. Smereka, *Semi-implicit level set methods for curvature and surface diffusion motion*, Special issue in honor of the sixtieth birthday of Stanley Osher, J. Sci. Comput., 19 (1-3), 439-456, (2003).
- [52] L. Vese, *A study in the BV space of a denoising-deblurring variational problem* Appl. Math. Optim., 44 (2), 131-161, (2001).
- [53] L. Vese, S. Osher, *Image Denoising and Decomposition with Total Variation Minimization and Oscillatory Functions*, J. Math. Imaging Vision, 20, pp.7-18, (2004)
- [54] J. Weickert, B.M. ter Haar Romeny, M. A. Viergever, *Efficient and reliable schemes for nonlinear diffusion filtering*, IEEE Transactions on Image Processing 3 (7), pp. 398-410, (1998).
- [55] T.P. Witelski, M. Bowen, *ADI schemes for higher-order nonlinear diffusion equations*, Appl. Numer. Math., 45, 331-351, (2003).